1

NINTH

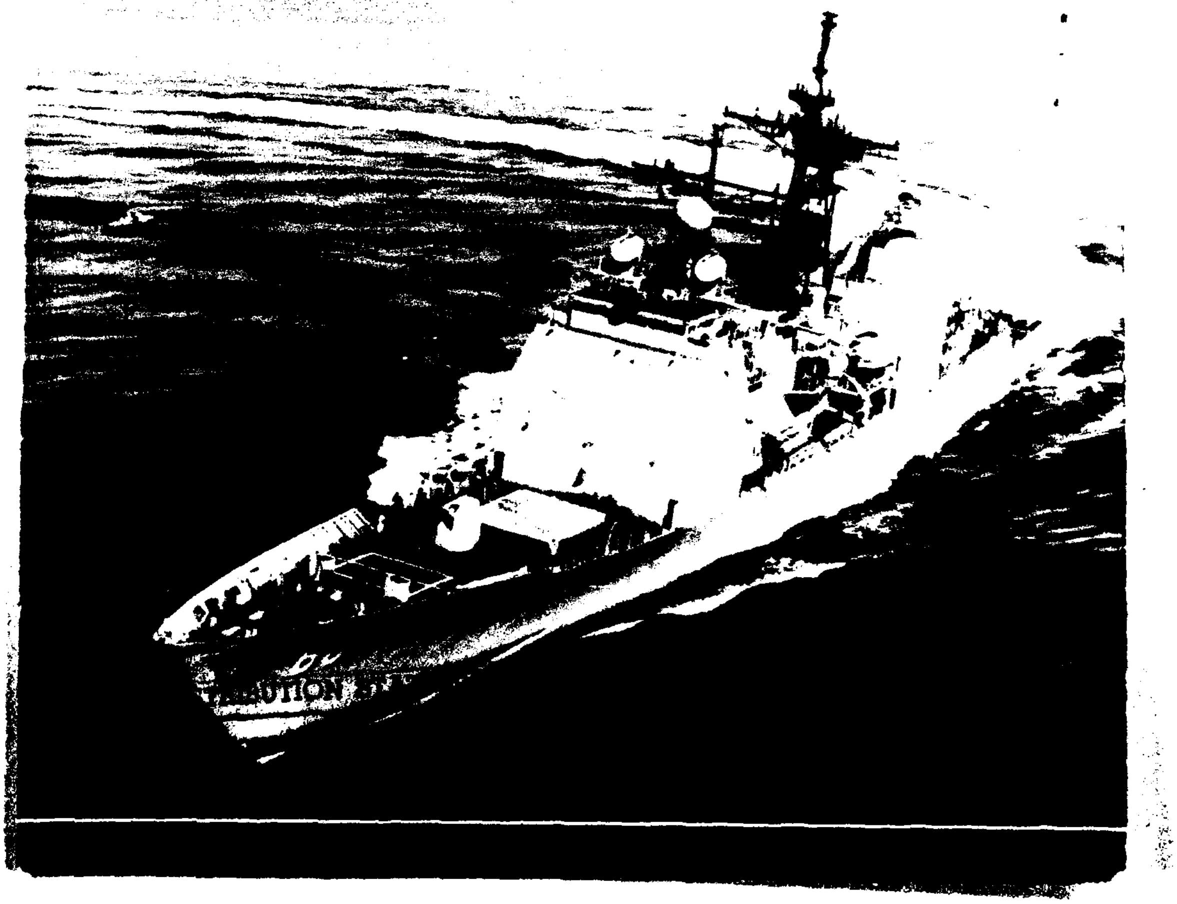
PROCEEDINGS

NINTH SHIP CONTROL SYSTEMS SYMPOSIUM

10–14 SEPTEMBER, 1990

BETHESDA, MARYLAND, U.S.A.

VOLUME 4



REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	14 SEP 90	PROCEEDINGS 10-14 SEP 90

4. TITLE AND SUBTITLE
PROCEEDINGS - NINTH SHIP CONTROL SYSTEMS SYMPOSIUM
VOLUME 4 of 5

5. FUNDING NUMBERS
NONE

6. AUTHOR(S)
MULTIPLE AUTHORS

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NINTH SHIP CONTROL SYSTEMS SYMPOSIUM
P.O. BOX 16208
ARLINGTON, VIRGINIA 22215-1208
USA

8. PERFORMING ORGANIZATION REPORT NUMBER
NONE

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
COMMANDER
NAVAL SEA SYSTEMS COMMAND
SEA 56Z4
WASHINGTON, D.C. 20362-5101

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NONE

11. SUPPLEMENTARY NOTES
Theme-Automation in Surface Ship Control Systems, Today's Applications and Future Trends

12a. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
Machinery Monitoring and Control Systems: Systems for electric plant, turbocharged diesel, and nitrogen generator; new generation Remote Terminal Unit; microprocessor-based systems; non-intrusive diagnostic systems; and cost effective specifications.
Software and Expert Systems: ADA programming and reusable software; RAST MK III controls; and a maritime maneuvering piloting aid.
Damage Control: Platform combat survivability and future requirements.
Simulation: Object oriented design method, rapid automatic passive optical ranging, manoeuvering fast time simulation, interconsole communication, dynamic track control, controllable pitch propellers, gas turbine/cpp ship stopping performance, and minehunter ship control.
Platform Management Systems: A 5000 point system and shipboard readiness reporting systems.
Human Factors: Vessel traffic modeling and a generic SCC analysis.

14. SUBJECT TERMS
AUTOMATION, SHIP CONTROL, CONTROL, MONITORING, ADA, EXPERT SYSTEMS, DAMAGE CONTROL, SIMULATION, PLATFORM MANAGEMENT, HUMAN FACTORS.

15. NUMBER OF PAGES
500

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

PUBLICATION INFORMATION

These papers were printed just as received from the authors in order to ensure their availability for the Symposium. Statements and opinions contained therein are those of the authors and are not to be construed as official or reflecting the views of the Naval Sea Systems Command.

Request for further information regarding the Proceedings and the Symposium should be addressed to:

COMMANDER, NAVAL SEA SYSTEMS COMMAND
DEPARTMENT OF THE NAVY
(ATTN: CODE 56Z4)
WASHINGTON, D.C. 20362-5101

VOLUME 4

TABLE OF CONTENTS

VOLUME 4

TABLE OF CONTENTS

VOLUME 4

TABLE OF CONTENTS

CONTROL AND MONITORING OF THE ELECTRIC PLANT ON THE CANADIAN PATROL FRIGATE

by Mr. N. G. Swamy, Mr. N. Bura
and Mr. W. Reinhardt
Department of National Defence, Canada

1. ABSTRACT

The Canadian Patrol Frigate (CPF) program will provide the Canadian Navy with twelve modern warships. As with the other systems in these ships, the electric plant incorporates control and monitoring features which enhance the availability of the plant while minimizing the need for continuous watchkeeping.

The generating plant consists of two pairs of diesel generator sets, one pair located forward and the other aft, with each pair connected to an associated switchboard.

In normal operation,the plant is controlled and monitored from the Machinery Control Room using the Integrated Machinery Control System (IMCS). Computer systems in each switchboard provide automatic control and the interface between the electric plant and the IMCS. Features such as automatic start and paralleling of the generator sets are provided to minimize the need for operator action under normal conditions. Comprehensive monitoring of the electric plant and well-designed man-machine interfaces further ensure that faults are rapidly diagnosed.

The reliability and availability of the electric plant is also enhanced by the provision of reversionary control of each pair of generator sets to the associated switchboard and the provision of manual operating modes. These provisions permit the electric plant to be operated despite failures in the automatic systems, or loss of control from the IMCS.

2. INTRODUCTION

Some twenty years separate the Canadian Patrol Frigates and the DDH 280 class destroyers, the previous batch of Canadian warships. In terms of the generating capacity and the broad aspects of power system design, the two classes have much in common. For example, the DDH 280 class ships have 3 x 750 kW gas turbine generators and one 500 kW diesel generator (replaced by a 1000 kW

diesel generator in the update underway at present), while the CPFs have four 850 kW diesel generator sets.

However, reflecting the many changes that have occurred over these years,the CPFs incorporate much more sensing and automatic surveillance.

The description of the control and monitoring system of the CPF electric plant in this paper provides a resume of the recent trends in the design of control and monitoring of naval power systems.

3. DESCRIPTION OF ELECTRIC PLANT

The 850 kW diesel generator sets, with their local control panels, are almost identical to the 750 kW sets used in the F 122 frigates of the German Navy. Due to this similarity, no qualification tests were required, however, each generator set and local control panel is subjected to routine inplant testing.

The main switchboards, being custom designed, were subjected to the full requirements of qualification testing, with a distribution section and a control section being subjected to shock tests. Routine factory tests on each switchboard include verification of all the metering and controls with the aid of a simulator, which provides the appropriate signals representing the generator sets and shore power.

The customary set-to-work and shipboard trials, which have commenced on the first ship, will verify the performance of the control and monitoring systems with all components interconnected.

Figure 1 is a schematic of the electric plant showing the main components of the system. The features of the plant are as follows:

a. The ship's maximum electrical load can be met by any two of the four generator sets. In accordance with naval practice, the sets and associated switchboards are located in well-separated forward and aft machinery spaces to enhance survivability of the electric plant under battle conditions;

b. A ring-main interconnection of the two switchboards provides a redundant bus tie between the switchboards;

c. Normal and alternate feeders from the two switchboards, routed through automatic or manually controlled transfer switches,

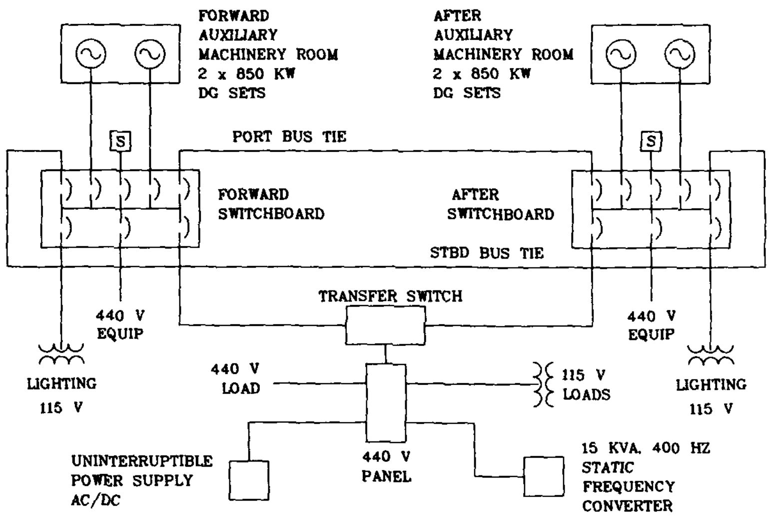


FIGURE 1 ELECTRIC PLANT

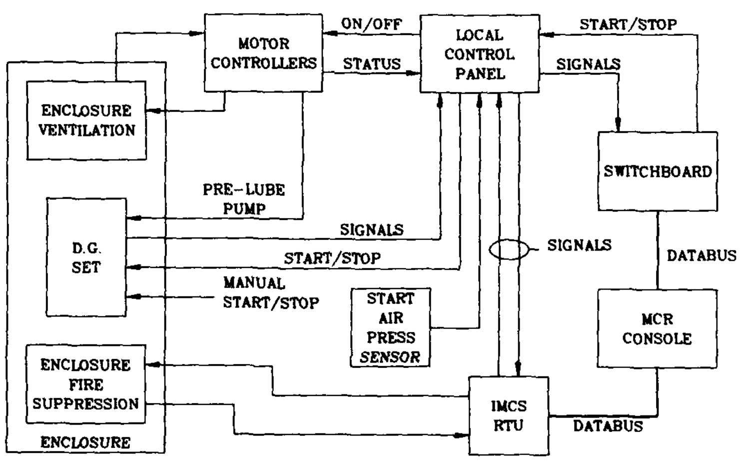


FIGURE 2 DG SET MONITORING & CONTROL

are provided for the power distribution panels feeding essential loads, and

d. A number of transformers, frequency converters and uninterruptible power supplies furnish 115 V, 60 Hz and 400 Hz power to loads located in their vicinity. This departure from the previous practice of having central power conversion and ship-wide 115 V and 400 Hz distribution systems reduces the weight of distribution system cabling, minimizes the penetration of watertight bulkheads and facilitates modular ship construction.

4. CONTROL AND MONITORING OVERVIEW

The above brief description of the electric plant indicates the particular attention given to survivability and reliability in its design. Consistent with this, the provisions for control and monitoring of the electric plant also include considerable redundancy.

Under normal operating conditions, the electric plant is operated from one of the consoles in the ship's Machinery Control Room (MCR) using the Integrated Machinery Control System (IMCS). Reversionary control at the switchboards allows control of the associated electric plant in the event of disruption of the IMCS or malfunction of the switchboard's computer system.

Automatic functions (ie. watchkeeper action not required) of the electric plant include:

a. The automatic start of one or more generator sets, designated as 'standby' by the operator, when the load on the generator sets in operation exceeds 80% of their combined rating. Due to the possibility of on/off cycling, shutting down of the additional set(s) is left to the watchkeeper;

b. Upon loss of power (black ship condition), a start signal is sent to all generators which are in 'standby' mode and the first one to come up to speed and voltage is automatically connected to its switchboard. The other sets remain running, available for the watchkeeper to bring on load;

c. Automatic tripping of one of the bus tie circuit breakers if three generator sets are put on line, and of both bus ties if four generators are put on. This is necessary due to the magnitude of the short circuit current available under the noted conditions. This infrequent occurrence of all generators running would be expected only during battle conditions;

d. Disabling of the automatic ship/shore paralleling procedure if more than one generator is in operation, and

e. Automatic shedding of non-essential loads when overload of a generator set occurs.

5. CONTROL AND MONITORING OF D.G. SETS

The 850 kW diesel generator set consists of a 16 cylinder, turbo-charged, air-start diesel driving an 1800 rpm watercooled generator. Each generator set is mounted on a raft which includes an acoustic enclosure.

5.1 Control Modes

A block diagram of the comprehensive monitoring and control facilities for the D.G. sets is shown in Figure 2. The following operational modes are available to minimize the effect of monitoring and control system disruptions on generating capability:

a. Remote control from the MCR with complete monitoring of all D.G. sets;

b. Remote control of a pair of sets from the associated switchboard. Diesel alarms/warnings and relevant generator parameters are displayed and full control of these sets is available;

c. Electrical start/stop and surveillance of the diesel at the Local Control Panel (LCP) mounted adjacent to the set;

d. Manual start of the diesel by activation of the ir starter;

e. Loss of control voltage is alarmed but does not cause shutdown of a generator set (control power is fed from uninterruptible power supplies and loss of control voltage would principally occur due to the failure of d.c. to d.c. converters within the diesel LCP), and

f. For test or maintenance on the D.G. set, a mode selector switch on the LCP permits the selection of one of four options:

-locked, in which all diesel starts are inhibited;

-local, in which remote start/stop is inhibited;

-test, in which remote start/stop is inhibited, plus closure of the generator circuit breaker in the switchboard is inhibited. In the Test mode, overspeed can be

simulated to verify proper operation of the protection against overspeed, and

-remote, in which local start is inhibited.

5.2 Starting and Stopping of the Set

Automatic pre-lubrication of the engine occurs as the first step in a normal start sequence. Emergency start, selected at the LCP by operation of a push button, or selected automatically when a generator set is required to start upon a power failure, a lows the engine to be run up without this pre-lubrication.

In a normal stop sequence, the engine is operated at no-load for a period and ventilation of the acoustic enclosure is maintained for some time after the engine has stopped. An emergency stop, in which the generator circuit breaker is tripped and the engine is stopped immediately (by the closing of flaps in the combustion air path), is initiated if any of the following conditions occur: high fresh cooling water temperature, low lub oil pressure, overspeed or fire in the set's enclosure. In the last case, ventilation of the enclosure is automatically shutdown.

5.3 Local Control Panel

A microprocessor based system is used for the sequenced start/stop and monitoring of the diesel. The 8-bit processor is equipped with 2K RAM memory and is capable of addressing 7x2K of EPROM, though only 3x2K are programmed in this application. Figure 3 is a simplified block diagram showing the interfaces between the micro-processor and the sensors and actuators on the engine.

Analog signals (variable resistances or milliamps proportional to temperature and pressure of lub oil, temperature of fresh water, thermocouple signal of exhaust gas temperature) are conditioned and compared with set points in five identical Printed Circuit Boards (PCBs). A different PCB provides the same function for the engine speed measured by a photo pick-up.

Digital signals are routed to the micro-processor through digital input PCBs. These PCBs, two in number, contain level translation, buffer and steering circuits. A set of 8 data signals, selected under program control, appears on the 8-bit data bus output of these PCBs. Digital outputs from the microprocessor are similarly processed through two digital output PCBs. Sufficient spare capacity exists for the addition of other features in the future.

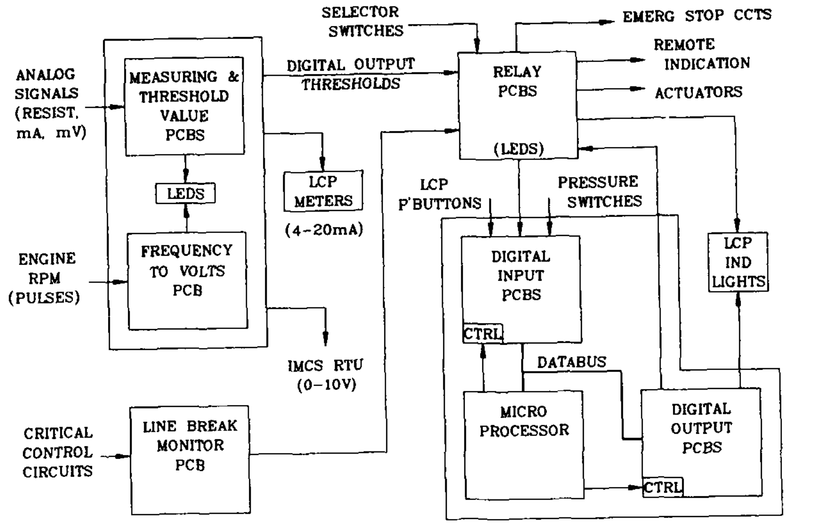


FIGURE 3 DIESEL ELECTRONIC CONTROL SYSTEM

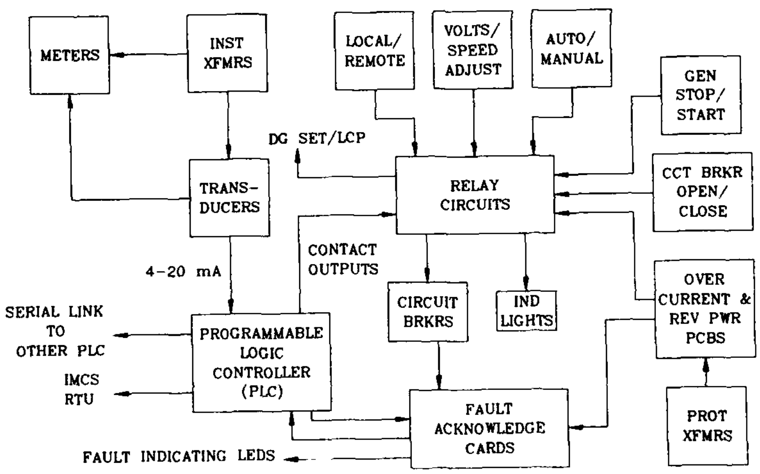


FIGURE 4 CONTROL & MONITORING AT SWBD

Six relay PCBs, each providing 6 channels, interposed between the digital input and output PCBs and the sensors, actuators etc. provide the auxiliary contacts, required for interlocking or latching, although some contact inputs from switches or outputs to indicator lights go directly to the digital input/output PCB. It should be noted that diesel emergency stop circuits are not routed through the microprocessor.

The monitoring arrangements for the D.G. set contain features that enhance reliability of the monitoring system and facilitate troubleshooting. Instruments located on the engine provide independent indication of temperatures and pressures which could be used to verify correct operation of the electronics and displays in the LCP. Essential circuits (stop solenoid, low oil pressure sensing, emergency stop flaps and speed sensing) are monitored for continuity and an alarm is signalled locally and remotely if a line break is detected. The interface PCBs are equipped with LEDs which indicate the status of a signal, whether a signal exceeds set thresholds or when a line break occurs.

5.4 Speed and Voltage Control

To avoid unnecessary complexity, the speed setting signal to the governor speed adjust motor and the voltage adjust rheostats are directly wired from the D.G. set to the switchboard. The generator's main and standby Automatic Voltage Regulators (AVRs) are mounted in a generator terminal box. Fault detection circuits in the AVR monitor the main AVR and, upon a failure, automatically transfer to the standby AVR. An alarm when this transfer occurs is hardwired to the switchboard and a test transfer is also incorporated in the switchboard.

6. CONTROL AND MONITORING AT SWITCHBOARD

As stated earlier, reversionary control of the electric plant is at each switchboard. The design of the CPF switchboards therefore incorporates all the features required for this control and also provides the facilities for remote control and monitoring from the MCR. Selection between local and remote (IMCS) control is made at the switchboard.

A battery-backed 28 V D.C. power supply is used for all the control and monitoring functions.

Figure 4 is a block diagram of the control and monitoring system for the electric plant.

6.1 Switchboard Programmable Logic Controller

The Programmable Logic Controller in each switchboard consists of a micro-processor section, data processing units, an interface unit and two data buses. The micro-processor is a Motorola 68000 unit, complete with various timer and coupler units, 32k ROM, 16k RAM and 16k EEPROM. The applications language used is Pascal.

Connected to the micro-processor, through Transmitter/Receiver data buses, are three digital input cards, each providing for 16 inputs, six output cards, each providing 8 Form C contacts, and two signal converter cards which convert 4-20 mA inputs from transducers to a voltage output for subsequent multiplexing and A/D conversion. The input cards take the various input signals, compare them to set-points and then send digital 'state' signals to the micro-processor where action is initiated. These cards are programmed in Assembly language.

An RS-422 serial link between each PLC, and between each PLC and the IMCS is controlled by the PLC.

Data passed between the PLCs is: status of the generator and bus tie circuit breakers, bus tie trip initiation or trip command, loss of voltage on bus signal, generator load, and synchronizing operations. In addition, initialization and PLC failure contact outputs are hard-wired between the two switchboards.

The control outputs from the PLCs are routed through electro-mechanical relays which provide additional contacts and a link with the backup manual controls. Each switchboard also contains a set of PCBs for the control of circuit breakers. The PCBs performing a protective function are the Overcurrent and Reverse Power PCBs. These PCBs provide contact outputs to trip the circuit breakers, via their undervoltage coil. PCBs which control closing of circuit breakers are the Synchro-Check and Phase Rotation (for shore power) boards. The contact output of the synchro check card controls closing of the desired circuit breaker when the backup manual permissive paralleling mode is selected. The contact output of the phase rotation card is routed as a digital input to the PLC.

Per normal design practice, instrument transformers for monitoring are separate from those for protection. However, the same transformers are used for the switchboard meters as for the transducers that provide a 4-20 mA output to the PLC. Manual controls and instrumentation on the switchboard are fairly conventional.

6.2 Control of Distribution Circuits

As shown in Figure 1, power panels supplying essential loads are fed from both switchboards via transfer switches. To avoid sudden transfer of all such loads to a generator, as would happen when a blackout is followed by the starting of a standby generator and its connection to the the bus bars, only the most essential loads are fed through automatic transfer switches. The other essential loads are fed through manual transfer switches.

As a new feature on Canadian ships, these manual transfer switches incorporate electrical controls similar to the automatic switches except that transfer is controlled by pushbuttons on the switchboards or through the IMCS. Status of both types of switches is indicated on the IMCS, with the status of the manual switches also indicated at the switchboards.

6.3 Fault Finding

Given the complexity of the switchboard controls, substantial aids to troubleshooting are a necessity and have been incorporated in the design. For example, automatic test of the PLC is supported by such aids as auxiliary contacts on digital output relays which are monitored by the PLC to verify relay operation, and a reference voltage along with a test channel on the analog/ digital converters. Two fault acknowledge cards provide local LED indication of particular fault conditions (e.g. undervoltage, overcurrent, reverse power) on generator and bus tie circuits and provide contact inputs representing these conditions to the PLC.

Loss of 28 V control power is indicated by a lamp on the switchboard and additional LEDs are provided within the switchboard to indicate the particular section in which power has been lost.

7. INTEGRATED MACHINERY CONTROL SYSTEM

The IMCS is a ship-wide distributed system which is designed to provide fault tolerant, highly reliable, centralized control and monitoring of marine systems and equipment.

7.1 Overview of the IMCS

The essential constituents of the IMCS are:

a. The Remote Terminal Units (RTU), dispersed throughout the vessel, which receive analog and contact inputs from machinery and sensors, process the data as required and then transmit it to other RTUs and the control consoles;

b. The control/display consoles in the MCR and the Bridge which provide the Man-Machine Interface and additional processing, and

c. The bi-directional, triplicated data bus. The communications protocol provides a high degree of message security with all stations polled every 300 milli-seconds. Data is transmitted on all three buses simultaneously, but received on only one. The receiving bus is continually changed to ensure that the system is fully operational.

The system is designed such that loss of any unit has a minimal effect on the remainder. With the processing spread throughout the ship, local operation of equipment will not be affected by damage elsewhere. Although loss of the consoles in the MCR will seriously disrupt control of most equipment, fallback control positions will allow continued operation.

7.2 Control and Monitoring by Watchkeepers

The watchkeeper on duty will call up the relevant page (our designation for the different screen displays available) by using either the keyboard, 'hot keys', trackball (similar to an upside down mouse) or software assigned keys. These software keys are a set of eight keys directly under the display whose function is assigned by whatever page is up.

The system being monitored/controlled will be displayed as a schematic diagram, showing data at the relevant monitor/control points, eg. voltages, circuit breaker status or generator status. If out-of-tolerance data, or faults occur, the display of the affected point changes colour (orange/warning or red/alarm) and shape (square to triangle) and flashes to attract the watchkeeper's attention.

If a problem occurs in a system not currently displayed, an alarm message is displayed at the bottom of the screen and remains there until acknowledged.

Taking as an example operation of the manual transfer switches, the watchkeeper first calls up the page which displays them. This will provide a listing of the switches, availability of normal or alternate power for each switch and the source to which the switch is connected. At the same time, the software keys are designated as 'Switch to Normal', 'Switch to Alternate' and 'Switch to other pages'. When the watchkeeper wishes to change the status of a switch, he/she uses the trackball to move the screen's cursor onto the switch's line, then presses the software key which

accomplishes the desired action. Assuming the switch transfers over, the display will then change to show its new status. If the switch does not transfer, after a set interval an alarm message will be displayed.

7.3 Control and Monitoring of the Electric Plant

The electric plant will normally be monitored and controlled from a console in the MCR. Although separate consoles are available to accommodate watchkeepers for propulsion, electrical and auxiliary systems, under normal conditions only one watchkeeper will be on duty at the central console.

The normal functions controlled by the watchkeeper are:

-designating sets as Standby #1, #2, etc. (for automatic startup);

-initiating the starting, stopping and paralleling of sets as required for rotation of running hours;

-initiating the paralleling, connection and disconnection of shore power;

The only automatic function done by the IMCS itself is the initiation of the start and automatic paralleling of a standby generator when the loading on the system reaches 80% of the on-line capacity.

Under action conditions, the watchkeeper would configure the system as required, ie. number of generators on line, bus-ties open or closed and possible rearrangement of normal/alternate feeds to the manual transfer switches.

Under damaged conditions, the watchkeeper will re-configure the system as far as possible in order to supply power to critical loads, through the manual transfer switches.

7.4 Monitoring Features

The status of the system can be assessed using the trend monitoring or data logging features of the IMCS. The operator can select various parameters, eg. kilowatt loading on the plant, with data recorded every 15 minutes over a period of time. This will permit an accurate assessment of the capability for future additions or changes to be made.

The health of the diesel generator sets can also be monitored, by logging temperatures and pressures for evaluation by

engineering staff. Future software enhancements may permit automatic assessment of engine health as a feature of the IMCS.

9. MAINTENANCE

The design features which simplify trouble shooting of the electronics have been previously described. Once a problem has been localized, test sets for the electronics in the LCP and the switchboard are available to test individual PCBs and to check the computer systems.

Depending on the predicted failure rates and their effect on equipment availability, test sets for specific boards may be carried on board. For the other boards, test sets will be held ashore and maintenance will be carried out by shore based (Second and Third Line) maintenance groups. It is intended to have PCBs repaired by the manufacturer.

10. CONCLUSION

The description of the control and monitoring facilities in this paper illustrates that although the modern shipboard electric plant is becoming more complex, enhanced survivability with reduced operator surveillance is possible through use of distributed computer based systems. Sufficient spare capacity and flexibility exists in the systems to accommodate future improvements.

Only operating experience on these ships will determine if the backup hard wired controls represent a transition to future all digital systems or should remain as essential insurance against 'fragile' electronic systems.

A NEW GENERATION OF RTU FOR MARINE CONTROL SYSTEMS

Marc Dietzway
and Michael L. Hagins
TANO Marine Systems, Inc.

1. ABSTRACT

This paper will discuss the new generation of Remote Terminal Unit that is being used on the Auxiliary Oiler T-AO 187 Class (198 Series). The paper will describe the latest developments in this new RTU technology including autonomous data acquisition and control modules as well as redundant power and communications busses within an RTU.

This paper will detail how this new generation of RTU will enhance marine monitoring and control systems by, adding versatility, increasing reliability, simplifying expansion, and greatly simplifying and cutting maintenance costs. It will also discuss the use of this same technology in other areas of marine monitoring and control systems.

2. INTRODUCTION

The T-AO 198 series of the T-AO 187 class of auxiliary oilers, which are being built by Avondale Industries, will employ a new generation of Remote Terminal Units (RTUs) built by TANO Marine Systems. This new generation of RTUs (known by the product name of TANOnet) has many advantages over earlier RTU designs including, autonomous data acquisition, added versatility, increased reliability, redundant power bussing, simplified expandability, and simplified maintenance.

Figure 1 depicts the older generation RTUs which required separate printed circuit cards for the signal input or output conductor interface (I/O), the signal processing electronics, and the data acquisition and communications computer. Usually, one card basket would be required to contain printed circuit modules suchas processors, memory controllers, Random Access Memory (RAM) modules, Read Only Memory (ROM) modules, serial line interface modules, voltage regulators, and Data Acquisition and Control (DAC) controllers. Another card basket would be required to house signal conditioners for the various types of input signals,

TYPICAL OLD GENERATION RTU

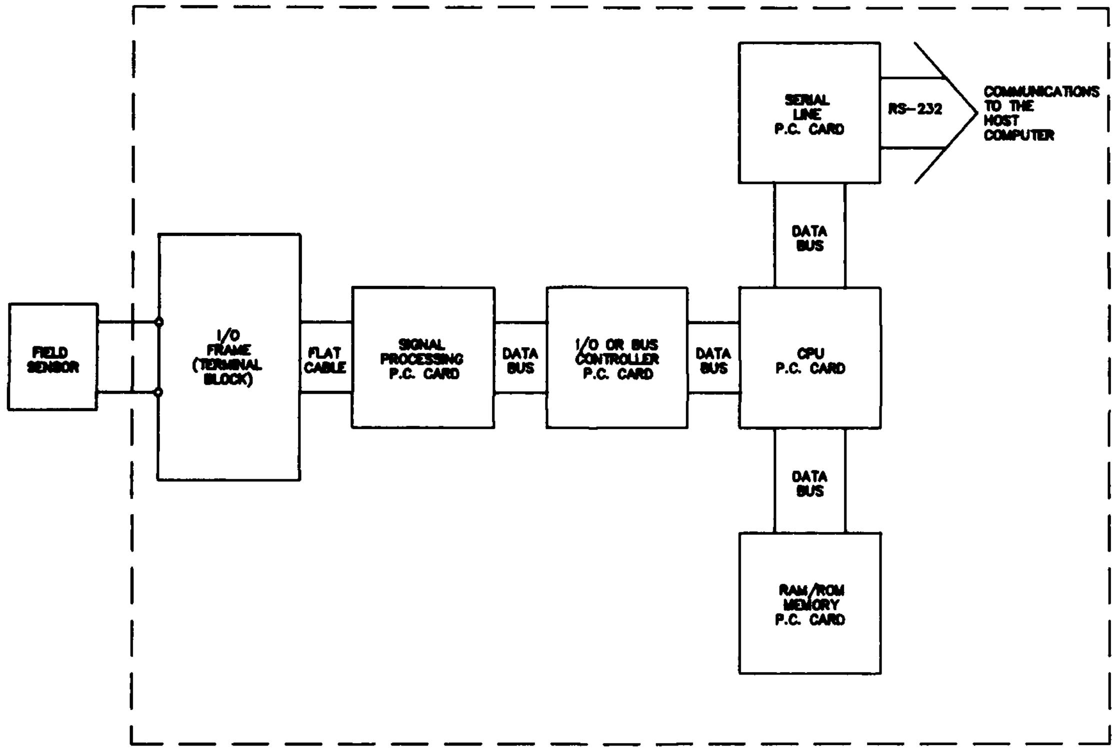


FIGURE 1

such as, 4 to 20 milliamperes, 0 to 10 volts, supervised contact, unsupervised contact, resistance temperature devices (RTD), thermocouple, contact output, analog output, etc. The I/O boards were normally standard EIA 19 inch rack mounted printed circuit cards which interfaced with the signal processing cards via either discrete conductors or flat cable.

Figure 2 depicts the new generation of RTU. These new generation RTUs do not use card baskets and separate cards for I/O, signal conditioning, and processing. A single, 19 inch rack mounted, card will perform all of these functions (for a particular set of input types), as well as generate its own special power requirements from an unregulated 24 volt supply. This means that RTUs can now be economically dispersed throughout a ship at locations closer to the associated equipment. This feature will save cable weight on new ships by reducing the distance between the end devices and the RTU. Several of the new RTU cards can be grouped together and controlled by a single processor card if the desired shipboard design so dictates.

3. DETAILS OF BENEFITS

3.1 Autonomous Data Acquisition

Each I/O module is completely independent and is a fully self-contained data acquisition unit, it has all the required signal conditioning, data converters, and an embedded microcontroller that performs all required I/O signal processing as well as performing internal diagnostics. These I/O modules perform as microRTUs in that they communicate with the RTU processor over a simple two-wire (RS-485) serial data link. In many applications today's RTUs do not require a processor module, the I/O modules can communicate directly with the host computer over this same RS-485 serial data link.

3.2 Added Versatility

Processor modules typically have a high-performance 16-bit (internal bus) microprocessor, a real-time clock, memory with battery backup, and may have as many as four (4) built-in serial communications ports. These modules can be used as an intelligent controller for distributed control applications. They also handle communications between its local I/O modules and a host computer through the use of one of the communications ports. The additional communications ports may be used for redundant communications to a host computer, local display, data entry/programming terminal or data logger.

Depending on the type of data points being monitored or

THE NEW GENERATION TANOnet RTU

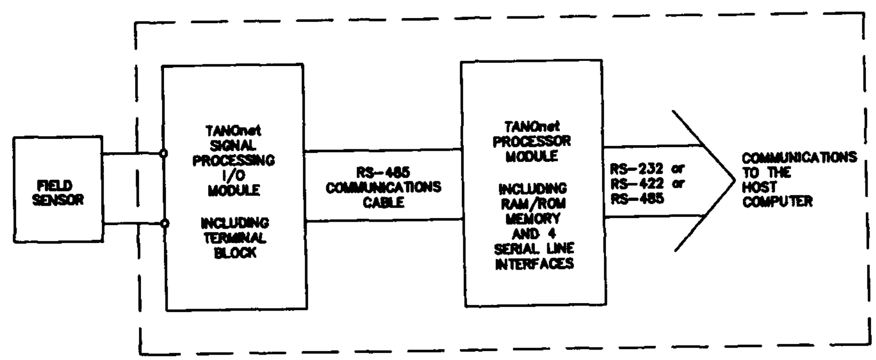


FIGURE 2

controlled, individual I/O modules can be channel by channel software configured. For example, on a relay output module, relays can be individually software configured to momentary, latching or duty cycled modes of operation.

3.3 Increased Reliability

Because the New Generation RTU processor consists of only one module, there are far fewer overhead electronic components in the RTU. By eliminating card baskets, there are no fragile wire wrapped back planes or mother boards. By performing the signal processing on the I/O modules, the use of flat cable is virtually eliminated. The elimination of the RTU level of voltage regulators and the use of power efficient CMOS technology, allows the RTU to run much cooler. All of these design considerations contribute to a more rugged and more reliable RTU.

3.4 Redundant Power Bussing

Each processor and I/O module has its own independent voltage regulators that require only unregulated +24 Vdc input power which is redundant and can be fused separately. The RTU's low-power consumption makes them ideally suited for battery and solar powered applications.

3.5 Redundant Communications Bussing

The TANOnet processor and I/O modules communicate over a common high-speed, 2-wire, RS-485 serial communications bus. These are the only common signal lines among the TANOnet I/O modules. Common dependent parallel data, address and signal buses are not utilized within the TANOnet system architecture.

3.6 Simplifying Expansion

Because there are no card baskets to limit expansion space, and very little internal cabling is used, expanding the data point capacity of an RTU is simply a matter of mounting a new I/O module in any available rail space, setting the I/O module's address and plugging in the combined power and communication cable. Of course the host computer's database must support these additional data points.

3.7 Simplifying Maintenance

Any I/O module(s) can be replaced with power on, without disrupting the rest of the RTU. The RTU will continue to operate properly with I/O module(s) missing, excluding the data from the missing I/O module(s).

There is no need to disconnect sensor wires from the terminals blocks in the RTU when replacing an I/O module. All I/O modules have heavy-duty, depluggable terminal blocks with a protective flip-top safety cover.

3.8 Built-in Firmware Diagnostics

For reliability and maintainability, all modules have a built-in watchdog timer, fault LED and status indicators. Each I/O module has its own internal diagnostics that run continuously and independent from other I/Os and processors. Each I/O frame type has different diagnostics specifically designed to evaluate the performance of the particular type of hardware resources onboard.

3.9 Low Maintenance Cost

Troubleshooting is as simple as identifying the I/O module on which the failed input(s) is on. This is best accomplished by looking-up the I/O module address of the failed data point available in the host computer database. Health LEDs labeled "RUN" and "TRANSMIT" are on all modules which also aids in the troubleshooting process.

Because the RTU's processor is on one module and all signal processing is accomplished on directly on the I/O module, there are fewer types of P.C. cards in the RTU and spares inventories are greatly reduced.

NEW MICROPROCESSOR-BASED MACHINERY SURVEILLANCE SYSTEM FOR TYPE 22 FRIGATES

by W. N. Pym
Racal Marine Electronics Limited
and M. K. Paddock
Sea Systems Controllerate
Bath, U.K.

1. ABSTRACT

HMS BROADSWORD, lead ship of the RN Type 22 class frigates, was laid down in February 1975 and commissioned in May 1979. The machinery surveillance system fitted in this ship, and subsequently her later sister ships, was adapted from a successful commercial marine system of that period, Decca ISIS 300.

Electronic technology has advanced rapidly since that time. This, together with the 10 year interval between the commissioning of HMS BROADSWORD and the last Type 22, HMS CHATHAM, faced the MOD Procurement Executive with the problems and increasing costs of supporting 'seventies generation electronics well into the 21st century.

This paper describes the requirements laid down to select a new system and to contain the costs of equipment changes. Features of the new microprocessor-based system are discussed and operational experience is reported, together with recommendations for future enhancements.

In conclusion, this paper reports on recent developments in the application of this system to commercial vessels.

2. INTRODUCTION

ISIS 300, introduced in 1969, and installed in nearly 50 NATO warships (in addition to several hundred commercial vessels), is a distributed, solid-state scanning system.

The ISIS 300 system configuration in the Type 22 (see Figure 1) has six local scanners, bulkhead mounted adjacent to the machinery under surveillance. Each scanner contains up to 40 single channel plug-in input acceptor modules. Analogue modules are equipped with high and/or low alarm setpoint potentiometers and provide 31 common measurement ranges for transducers calibrated to give an input signal of 0-100mv DC for zero to full scale. Alternative input acceptor modules are supplied for PRT's (RTD's) and switch type inputs.

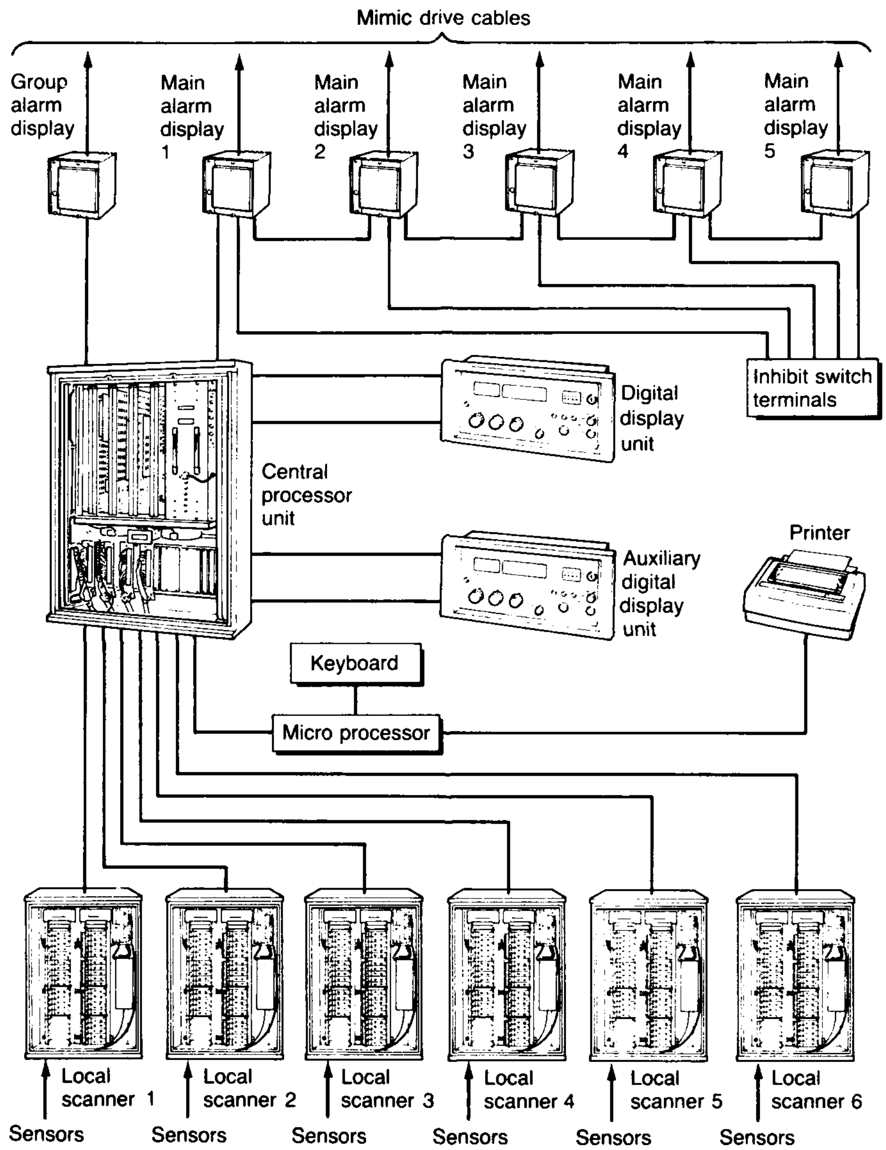


Figure 1 ISIS 300 system configuration

Transducer signals are amplified within the scanners and the channel data from each scanner is transmitted sequentially through large multicore cables to the central processor, which contains the bulk of the system logic on eight large printed circuit boards.

Two digital display units are provided. The primary unit is located in the Machinery and Electrical Control Console Assembly (MECCA) with an auxiliary unit in the EOOW's desk. Each unit contains three decade switches for channel selection, with the number of the selected channel displayed on the first three of eight 'Nixie' digital indicator tubes. The remaining indicators display a four digit value with minus sign when appropriate. Display mode switches allow the presentation of channel value in the appropriate engineering units, high and low alarm limits, system time or test values.

The central processor also provides outputs to a group alarm display and five 40 channel main alarm display drivers. These latter units are connected in turn to individual alarm lamps in the MECCA mimics.

The remaining major item of equipment is a printer for data logging and alarm history recording.

While the above outline is necessarily brief, ISIS 300 was a leader in its field in the 'seventies with many innovative features, including the distinction of being the first distributed, solid-state scanning system designed to marine specifications.

More recently enhancements have been made to ISIS 300, the most significant of which is the improvement in the Man-Machine Interface (MMI) by the addition of a VDU at the watchkeeping position. This releases the watchkeeper from the laborious task of dialling up channels on the digital display, allowing him to select important alarm channels and display them constantly throughout his watch.

Limitations and cost considerations were taken into account in the improvement of a system that employs outdated technology, but the worrying factor remained the creeping obsolescence and consequent inevitable decline in the reliability of the system.

Electronic technology has advanced rapidly in the last ten years. When viewing ISIS 300 from the perspective of today's technology, one is immediately struck by its massive construction: e.g. scanners (Figure 2) with dimensions 1090 x 540 x 250mm, weighing nearly 80kgs each. The restricted and time consuming MMI, with its single line Nixie tube display accessed through multiple decade and mode switches, becomes immediately apparent. These shortcomings, however, are overshadowed when one considers the 10 year interval between the commissioning of HMS BROADSWORD and the last Type 22, HMS CHATHAM.

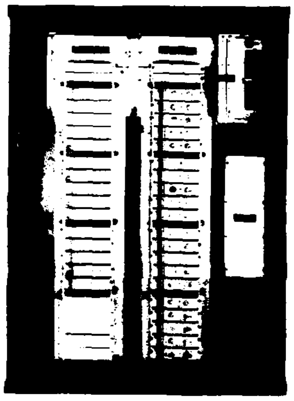


Figure 2 ISIS 300 local scanner

3. SYSTEM REPLACEMENT CRITERIA

The obvious solution was to replace ISIS 300 with a new and more cost-effective microprocessor-based system and in 1985 the Procurement Executive commissioned an independent market survey to find a system that could meet the following criteria:

- As a minimum, provide the function and facilities of the current ISIS 300 Machinery Surveillance System.
- Use existing technology to the fullest extent i.e. minimising 'high risk' development.
- Be compatible with existing transducers, space restrictions, mechanical interfaces, transducer wiring, ship's cabling and other alarm and warning systems.

A list of likely suppliers, including reputable manufacturers with sufficient experience in non-marine fields as well as those with proven naval and/or rugged marine environment track records, was compiled. These manufacturers were invited to submit equipment design specifications for analysis under the following requirement:

- Overall equipment and installation costs.
- Enhanced MMI.
- Installation.
- Maintainability.
- Support.

3.1 Overall Equipment and Installation Costs.

a Cabling Althoug[illegible] [illegible] systems are usually cheaper than earlier generations, both in initi[illegible] [illegible] through life costs, a major concern was the expense (and considerable disturbance) of replacing system cabling, both transducer to surveillance system cabling and unit interconnection cabling. The selected system was to utilise existing cabling wherever viable, particularly where watertight bulkhead penetration between compartments was involved.

b Transducers For any major surveillance system, a large percentage of the total cost is made up by the on-plant instrumentation, i.e. pressure and temperature transducers etc. It was therefore a requirement that the new system would utilise, wherever possible, existing transducers, both analogue and digital. In the case of analogue transducers with 0-100 mV DC signals, provision was to be made to permit future conversion to the more widely accepted standard 4-20 mA DC, with minimum disruption to the existing system.

c *Existing Alarm & Warning Systems* The selected system was to be capable of integrating fully with existing alarm and warnings as employed in the MECCA, Gas Turbine and Gearbox Vibration Monitoring Systems.

3.2. Enhanced MMI

A means of improving the existing display of alarm and warning data was an important consideration in the new system. The visual display unit had already proved to be a valuable improvement to the ISIS 300 system, and this was expected to form the MMI within the new system. It was considered that the introduction of this form of MMI would reduce watchkeeper workload, facilitate the presentation of multi-channel surveillance information and, of no lesser importance, introduce a medium with which a watchkeeper would be familiar.

All system data was to be accessed from a user-friendly, keyboard-driven visual display unit, structured by menus, where the re-configuration of all channel data could be carried out if required under password control.

Features from the MMI were to include the following:

- Channel information displayed in tabular or bargraph form.
- Special logs and reports.
- Selected lists.
- Capability for both fixed and portable VDU operator stations independently accessing data from the system.
- Local scanner units capable of operating in an intelligent stand-alone mode with local display of alarm and warning parameters, with the facility for monitoring equipment locally during test, trials and fault-finding.
- Serial data links interfacing with other related systems i.e. local control panels, computers and modems.

3.3 Maintainability

The system was to be capable of comprehensive system self-checking, including indication of earth faults and sensor failures, with the availability of test functions for all lamps, keyboards, monitors and communication links.

3.4 Support

a *Spares* With the selected replacement system likely to be retrofitted in three major classes of warships, totalling some 25 vessels, special-to-purpose items were to be kept to an absolute

minimum. The use of a common range of modules distributed throughout the system and capable of being used across the three classes of vessels was considered essential.

b Equipment Life The selected system was to be supportable for a minimum of 20 years and make provision for advances in electronic technology.

c Training It was desirable that the selected system be capable of being re-configured for a shore-based training simulator, providing structured operator and maintainer training.

3.5. Installation

Retrofit of the new system was to be possible without dockyard assistance within a 10 day period, to provide the opportunity of accomplishing the work during a berthing period while the ship remained in commission. All remote system units were to fit within the space envelopes of the original equipment and be capable of accepting existing cable harnesses without major re-work.

4. THE SOLUTION

Detailed analysis of the market survey results revealed that the Racal-Decca microprocessor-based ISIS 250, 3rd generation successor to ISIS 300, was a prime candidate as a replacement system. This choice was confirmed after a comprehensive six months minor trial in the RN hydrographic survey vessel, HMS HECATE (the original trials ship for ISIS 300 in 1971). During this trial, HMS HECATE spent 85% of the time at sea with the system in full use. The equipment was retrofitted in three days using 90% of existing cabling and transducers, and was found to be very reliable, with considerable improvements made to the watchkeeping capabilities within the vessel.

The system on HMS HECATE, however, was small and in no way representative of the comprehensive system required for the surveillance of gas turbine propulsion, gearboxes and the wide range of auxiliary machinery found in a modern warship. A further minor trial was initiated to prove the retrofit of a full ISIS 250 system and its performance under operational conditions on a T22 frigate prior to class installation.

4.1 System Technical Description

Figure 3 illustrates the replacement surveillance system. All original cabling is retained, although significantly fewer cores are used in interconnecting the major units. Additional cabling is minimal, being limited to local connections between keyboards and monitors, etc.

Data from remote transducers is collected in 48-channel local scanning units (LSUs). Each LSU contains three 16-channel analogue/logic input modules, an interrogation panel and a processor module.

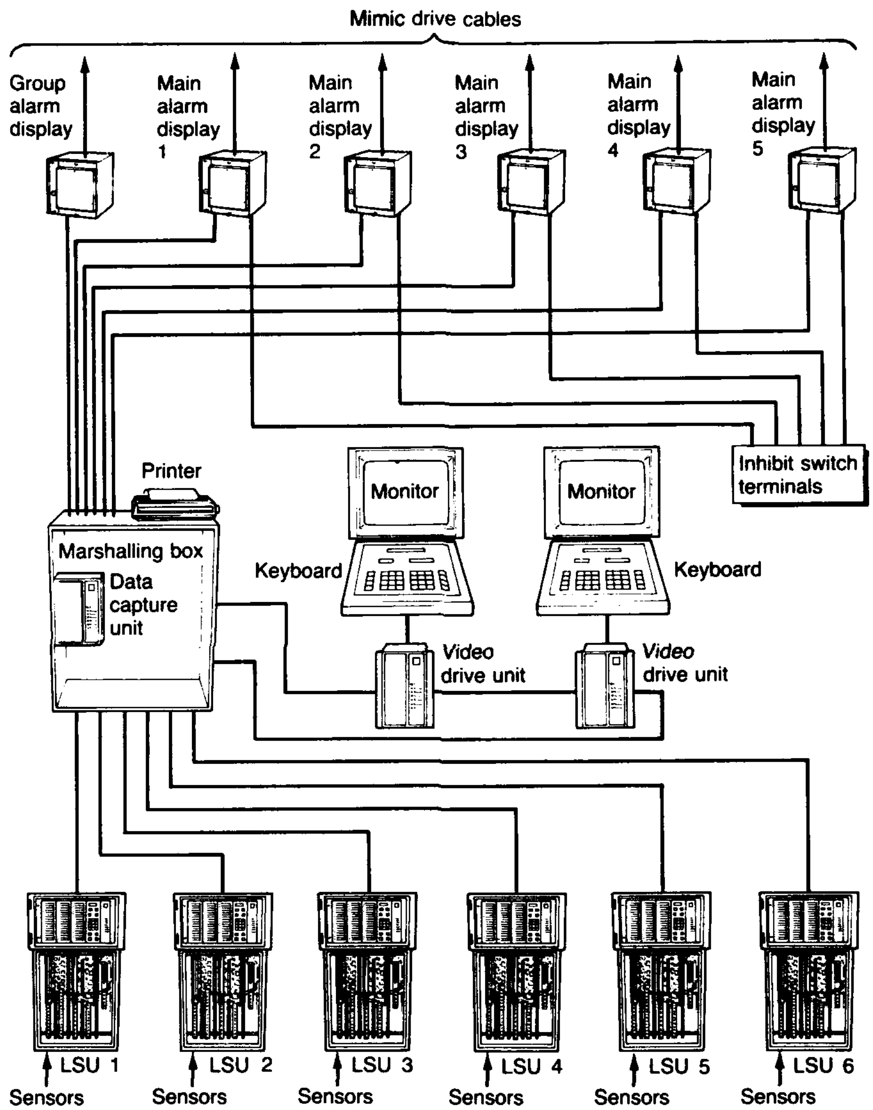


Figure 3 ISIS 250 system configuration

The input module will accept 4-20mA, 100 ohm PRT or contact signals wired directly, in any mix. All other signal types are handled by a series of small interface units that convert the incoming signals to 4-20mA. This technique is used extensively in HMS BROADSWORD. Referring to Figure 4, the lower portion of the LSU enclosure houses these interface units, positioned to replicate the locations of the original system cable terminations. A further benefit of this arrangement is that when the original transducer is later replaced by a 4-20mA equivalent, the interface unit can be readily removed and replaced by a simple jumper connection.

The interrogation panel permits the maintainer to read each channel's measurement value, high and low alarm setpoint locally, while the processor module updates all channel data every second for onward transmission to the central displays. The processor module also provides another important feature by ensuring that each LSU will continue to function autonomously in back-up mode automatically in the event of loss of communication with the central displays.

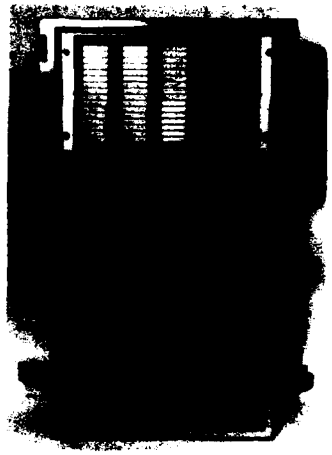


Figure 4 LSU enclosure

The data capture unit (DCU) in turn continuously polls all six LSUs via the RS 485 serial communication link with these units. The DCU, mounted in the marshalling box that replaces the CPU of the original system, consists of an identical processor to that used in the LSU, but with a different software program.

Two VDU operator workstations are provided in the MECCA and on the EOOW watchkeeper's desk respectively. Each workstation consists of a video drive unit, a multifunction alphanumeric keybord and a colour monitor. The video drive unit once again uses the same processor module as described earlier, together with a video processor module. This latter unit handles all video processing so that a wide range of conventional colour monitors can be used with the system, eliminating the need for costly and specialised visual display units.

As in the earlier system, outputs are provided to individual alarm lamps in the MECCA mimics. Once again, provision has been made to simplify the rapid retrofit of the earlier system. In this particular case, a special printed circuit board was designed that is compatible with the new ISIS 250 system, but which fits into the existing frame supplied with the original system, again without disturbing existing wiring.

A printer for data logging and alarm history recording, mounted on the marshalling box (with paper feed trays inside the box), is also supplied to complete the system. However, in common with the rest of the new system, this unit provides a far wider and more flexible range of operator and maintainer functions as described below.

Reference has been made to the repetitive use of the same processor *module in various units.* This is part of a deliberate policy to simplify system troubleshooting and spares provisioning. As a result of this approach, the principal printed circuit boards in the system have been reduced to four types. This also considerably simplifies the task of coping with future technology changes by printed circuit board update and replacement.

4.2 System Operation

Before proceeding to describe the operation of the system, it is important to emphasise that the ISIS 250 was designed to commercial marine specifications, like its predecessor, ISIS 300. These specifications are embodied in the rules and regulations of the major international classification societies for periodically unattended machinery spaces. The increasingly demanding environmental specifications, and harsh economics of the shrinking shipbuilding industry in the late 'eighties have all contributed to the evolution of this latest ISIS.

It was gratifying therefore to learn that the Ministry of Defence wished to take advantage of this hard won commercial experience and distill it to meet naval requirements; not to impose new and rigid specifications. The result of this co-operative approach was an extremely economical and cost effective system, whose operation is essentially identical to its commercial equivalent. ISIS 250 displays are organised in an easy-to-understand hierarchy, starting with a representation of the keyboard to guide the operator in accessing the primary level of displays, as shown in Figure 5.

Once any of the primary displays, e.g. latest alarms, group alarms, logs and reports etc., has been selected, information messages are provided at the bottom of the screen to advise the operator how to proceed down through the display hierachy.

a Latest Alarms Pressing the primary level ALARMS key will display the first page of latest alarms changes, with the most recent event at the top of the screen. A red background denotes an alarm and a green background a return to normal. Up to 3 pages of latest alarms are available on the screen, i.e. the latest 60 alarm events.

A new alarm will flash the relevant alarm status word. When the alarm is accepted, the status word will stop flashing and go steady. In addition to alarm status (such as high, low) channels inhibited whilst in alarm are also displayed.

A further valuable feature is that the system recognises out of range signals and displays FAIL to denote either a sensor failure or earth fault external to the ISIS equipment, with discrimination to a single channel.

An option is available during system configuration whereby a return to normal is displayed for 5 minutes only, after which time both the return to normal and the corresponding alarm entry are removed from the display. This feature ensures that only outstanding alarms and recently corrected alarm conditions are displayed. No information is lost if this option is selected as all

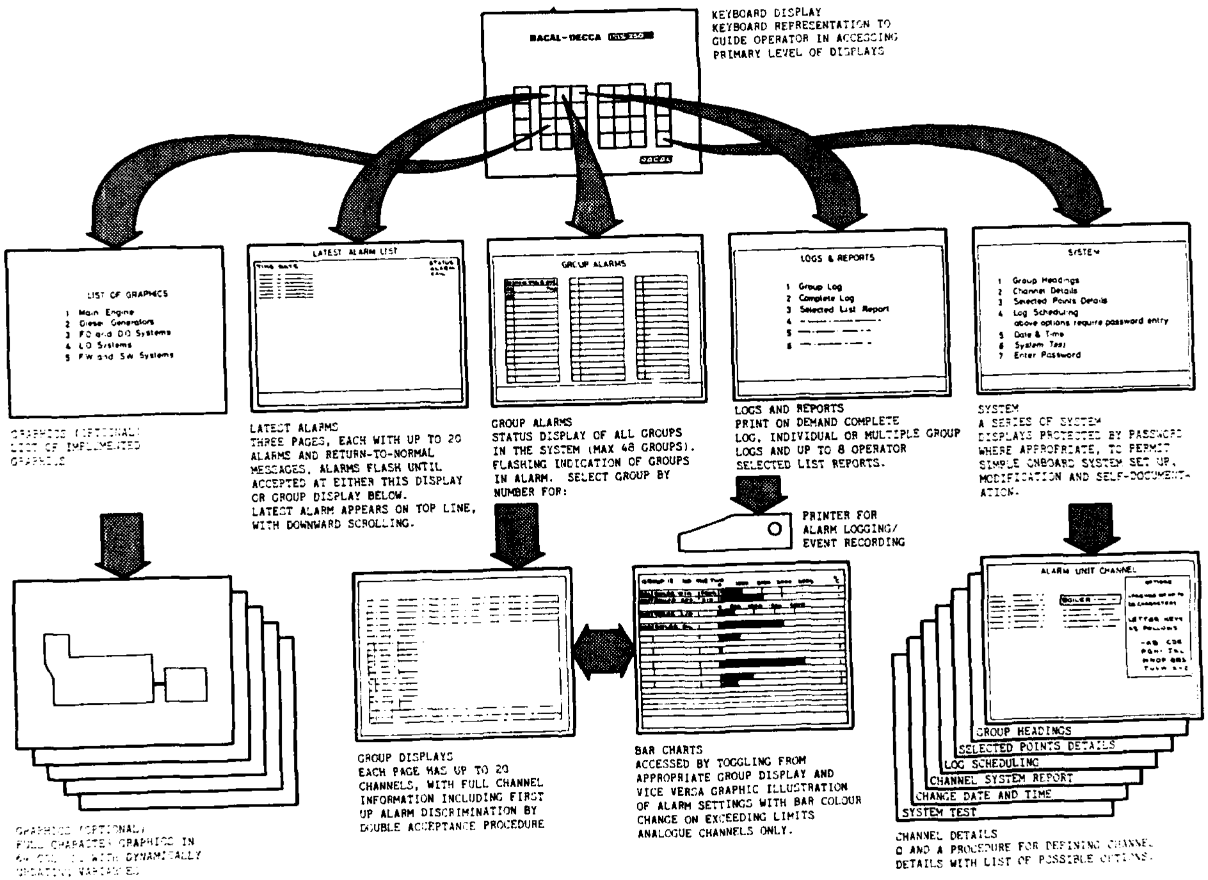


Figure 5 ISIS 250 display hierarchy

alarm events are printed automatically on the system printer with time and full details. Return to normal messages are also handled similarly.

b Group Displays and Bargraphs The second primary operating display is accessed through the GROUP button on the keyboard. This presents an overview of all the groups in the system. All input channels may be configured from the keyboard in up to a maximum of 48 groups. There is no limitation on the number of channels in any group. However, each channel can only be assigned to one group.

In addition to the 48 configurable groups, a special group, designated SYSTEM GROUP, continually monitors the health of all communications between each unit in the system.

To proceed down the hierarchy, the operator selects the group number of interest and is then presented with a tabular display of all the channels, both analogue and binary, in that group. If more than 20 channels are assigned to a group, additional pages are provided. The tabular display identifies channel number, legend, current value of input, measurement units, upper and lower alarm units and status.

A toggle button on the keyboard permits the operator to select an alternative horizontal bargraph presentation of all the analogue channel inputs in this group, as illustrated in Figure 6. The primary purpose of this display is to allow the operator to rapidly scan the condition of the analogue channels 'at-a-glance'. it should be recognised that there may be binary channels also configured in this group which will not be apparent from this particular display and therefore alarm acknowledgement *can only be effected* from the companion tabular display which lists *all* channels.

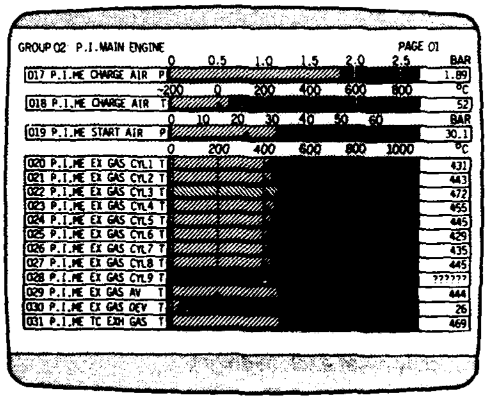


Figure 6 Bargraph display

In the bargraph presentation, each channel will occupy either one or two lines; consequently the number of channels on a page is no longer fixed. For this reason there will be no correspondence between the page contents in different modes. Two lines are normally required because the scale of the bargraph appears on the upper line and the bar itself appears on the lower line. Where several consecutive channels share the same scale, only the first channel takes two lines since the scale is not repeated unless it is different. The format from left to right is:

first line	scale markings measurement units
second line	channel number channel text bar graph channel value

The bargraph has arrowheads indicating low and high alarm limits and the colour is green for a channel which is not in alarm, yellow for a channel which is inhibited and red for a channel which is in alarm. No distinction is made between acknowledged and unacknowledged alarms in this mode.

c *Logs and Reports* Pressing the primary level button 'LOGS' acesses the menu of logs and reports, permitting the operator to select the following for printout on demand;

(a) Log of any group.

(b) Complete log.

(c) Single channel log.

(d) Selected list report.

If an alarm (or return) occurs while a log is in the course of printing, the alarm message will be stored until the log printout is complete and then printed with the actual time of the alarm occurrence. This ensures that a single printer can handle both logs and alarms without over-printing.

Scheduled logs are printed automatically and can be scheduled at any time interval from 1 minute to 24 hours. For logs that are scheduled at intervals that are not divisible into 24 hours, the schedule is automatically reset each midnight.

d Selected Lists The ISIS 250 selected lists feature provides the watchkeeper and maintainer with an extremely useful tool to construct his own temporary groups, bargraphs and printouts, without disturbing the main system displays. Accessed and constructed through the system administration function, there are eight lists available, each with up to 32 channels. A separate set of eight selected lists is available at each independent VDU workstation.

In each list, the operator can assign any channel throughout the system, combined in any order and then:

(a) Display the list in tabular form on demand.

(b) Display the list in bargraph form on demand.

(c) Where a printer is connected to the workstation, print the selected list at time intervals from 1 minute to 24 hours, or at any time on demand.

e System Administration Pressing the 'ACTION' button from the top level keyboard display provides access to a menu of system administration functions. At the discretion of the MEO, any or all of these functions can be protected from unauthorised entry by means of an alphanumeric password.

The various displays within this particular hierarchy permit the full configuration of the system, including text, from the keyboard. This feature eliminates the need to return equipment to the factory if last minute system or channel configuration changes are required.

Figure 7 shows a typical configuration page for a single channel. As the operator moves through a question and answer routine, the legal choices for each parameter are presented in a window on the right hand side of the screen.

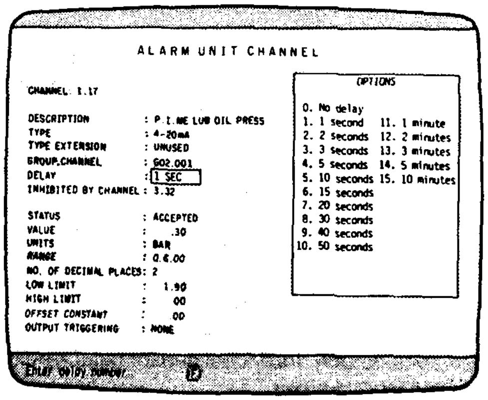


Figure 7 Typical configuration display

On completion of the channel configuration, another system administration function permits the operator to print out a fully documented copy of all channel details for record and checking purposes.

Other functions included in this section are:

(a) Change date and time.

(b) Function test of workstation and communications.

(c) Log scheduling.

(d) Changing selected list details.

(e) System and option configuration.

It should be noted that none of the displays illustrated in this paper reflect actual Type 22 operational parameters.

5. OPERATIONAL EXPERIENCE

The authors visited HMS BROADSWORD in early March 1990 to obtain first-hand reactions to the new system. It was immediately noted that, although there had been some major changes in the marine engineering staff since the ISIS 250 installation, a high degree of acceptance of the system was evident.

The watchkeepers have found the system easy to use and confidently demonstrated their mastery of the display hierarchy. Particular use had been made of the flexible selected list feature which had been well received. Examples of its use were:

- Engineer Officers of the Watch used this facility extensively, having their own personal lists showing parameters which they may, on certain occasions wish to monitor closely, adding or removing channels from the list as required.

- Special lists of tank levels were found helpful for fuel bunkering and transfer etc. Particular use was made of the bargraphs in this instance.

The maintainers were equally adept with the system and conversant with the system administration features which they found most helpful. They acknowledged that, as in all marine installations, the primary maintenance work was involved in checking transducers, particularly those in inaccessible areas. ISIS 250, with its local interrogation panels in each LSU, simplified this task considerably. Individual sensor failure indication and system health/fault-finding features were praised.

Perhaps the most surprising finding in summary was the confident matter-of-fact use of the system by the young watchkeepers, all of whom had been trained on board.

5.1 Recommendations for Future Enhancements

a *Interfacing with Local Control Panels* Investigations are in hand at the time of writing this paper to employ the existing trending and graphics packages, available as options with ISIS 250, on primary and secondary surveillance of diesel generators, without necessarily increasing the number of parameters measured directly by ISIS 250.

This can be achieved by a serial link (RS232 or similar) between the microprocessor-based local control and surveillance panel and ISIS 250, facilitating a 'dump' of data from parameters measured by the local control panel (e.g. lub oil pressure, exhaust gas temperature etc.) for analysis by the ISIS.

b *Graphics & Trending* The use of keyboard-configurable standard engine graphics would form a continuous dynamic representation of particular equipment parameters, e.g. for diesel generators, where a total picture of alarms and warnings would be invaluable in giving the running characteristics of an equipment following a major overhaul, repair or indeed new installation.

A secondary function of the graphics would allow the watchkeeper to familiarise himself with a transducer fit on a particular equipment or system, thus providing a useful on-board training facility. In addition, the system's trending package could be helpful in the short and long term condition monitoring of certain equipment following ovehaul and repairs, and in predicting future maintenance tasks.

c *Exhaust Gas Temperature Averaging (EGA)* A further potential enhancement to the system's condition monitoring capabilities is to use the EGA feature that is resident in all ISIS 250 systems. This feature uses dynamic limit setting, i.e. limits vary with engine load. As shown in Fig 8, tight deviation limits can be set at full load, with wider limits at engine idling temperatures. Below this point, the EGA facility is inhibited. In this manner, tight control of engine balance can be maintained.

d *Closed Circuit Television (CCTV)* To increase the flexibility of ISIS 250 for enhanced surveillance of unmanned machinery spaces, sensitive areas and for damage control scenarios, consideration is being given to integrating a CCTV element. The CCTV proposed would integrate with ISIS 250 in that the workstation colour monitor would be switched between a selected camera and alarm and warning data. This would optimise the use of display space in the ship control centre.

When a camera is selected a common pan and tilt joystick would control directions, whilst a common zoom 'in'/'out' push switch controls the lens field of the selected unit.

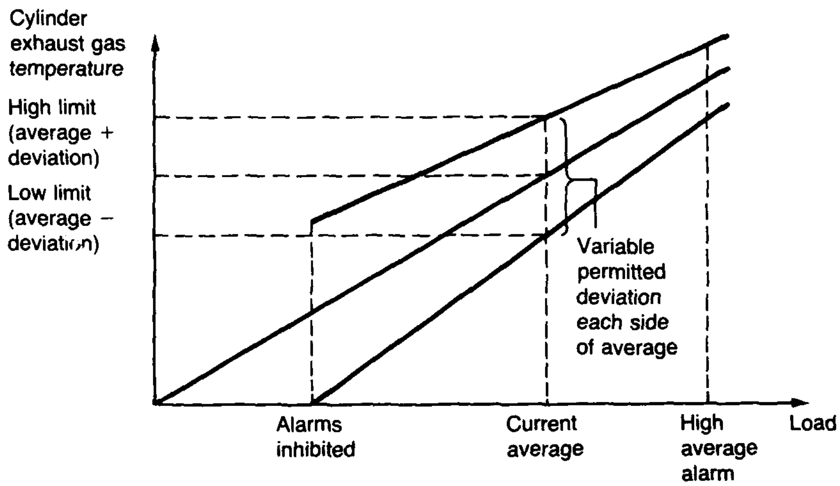


Figure 8 Deviation from average monitoring (dynamic limits)

An override switch would automatically return the display to ISIS in the event of an alarm condition being detected.

6. COMMERCIAL DEVELOPMENTS

Reference has been made earlier to the commercial evolution of ISIS 250. This process continues in response to market demands. Briefly outlined below are some further facilities that have been added in the last two years (in addition to the graphics and trending mentioned earlier) which may also have potential naval application:

6.1 Bearing Temperature Monitoring

Experience has shown that main bearing problems in marine diesel engines can develop rapidly, often with catastrophic results. Early indication of an abnormal condition is therefore highly desirable.

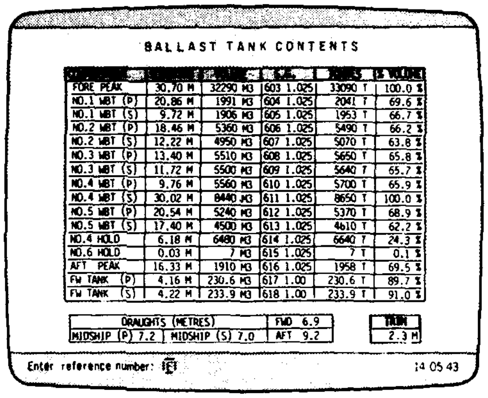
While high temperature alarms are normally provided for each main bearing, further security can be obtained by adding channels that provide alarms (often coupled with an engine slowdown output) for excessive rate-of-rise in bearing temperatures. This can often provide an early warning, even before the corresponding high temperature alarm limit is reached.

This feature is available as an option with ISIS 250. Parallel channels, i.e. two per bearing, provide both absolute temperature and excessive temperature rate-of-rise alarms. Each temperature is sampled once per second and the rate-of-rise function is enabled once the bearing reaches its preset normal operating temperature.

An alarm, is generated on detection of a rise in temperature of more than 2°C (adjustable) in a preset time.

6.2 Tank Contents Calculations

A typical tabular display of tank contents is illustrated in Figure 9. In this instance, ballast and FW tanks are summarised, although this facility applies equally to other tank contents such as fuel oil, etc.



BALLAST TANK CONTENTS

[illegible]	[illegible]	[illegible]	S.G.	TONNES	% VOLUME
FORE PEAK	30.70 M	32290 M3	603 1.025	33090 T	100.0 %
NO.1 WBT (P)	20.86 M	1991 M3	604 1.025	2041 T	69.6 %
NO.1 WBT (S)	9.72 M	1906 M3	605 1.025	1953 T	66.7 %
NO.2 WBT (P)	18.46 M	5360 M3	606 1.025	5490 T	66.2 %
NO.2 WBT (S)	12.22 M	4950 M3	607 1.025	5070 T	63.8 %
NO.3 WBT (P)	13.40 M	5510 M3	608 1.025	5650 T	65.8 %
NO.3 WBT (S)	11.72 M	5500 M3	609 1.025	5640 T	65.7 %
NO.4 WBT (P)	9.76 M	5560 M3	610 1.025	5700 T	65.9 %
NO.4 WBT (S)	30.02 M	8440 M3	611 1.025	8650 T	100.0 %
NO.5 WBT (P)	20.54 M	5240 M3	612 1.025	5370 T	68.9 %
NO.5 WBT (S)	17.40 M	4500 M3	613 1.025	4610 T	62.2 %
NO.4 HOLD	6.18 M	6480 M3	614 1.025	6640 T	24.3 %
NO.6 HOLD	0.03 M	7 M3	615 1.025	7 T	0.1 %
AFT PEAK	16.33 M	1910 M3	616 1.025	1958 T	69.5 %
FW TANK (P)	4.16 M	230.6 M3	617 1.00	230.6 T	89.7 %
FW TANK (S)	4.22 M	233.9 M3	618 1.00	233.9 T	91.0 %

DRAUGHTS (METRES)		FWD 6.9
MIDSHIP (P) 7.2	MIDSHIP (S) 7.0	AFT 9.2

TRIM
2.3 M

Enter reference number: 14 05 43

Figure 9 Tank contents display

Standard level displays are catered for in the system in a similar manner to other analogue inputs, such as pressure, temperature, etc. That is, level information is provided in both tabular and bargraph form as 0-100 percent level, with high and/or low level alarms as appropriate.

An additional DCU, identified as a sequence processor, is used when displays of capacity and/or mass are required. To achieve this, tank table data of level vs capacity, tonnes, barrels etc. is stored in look-up tables held in the sequence processor software. Specific gravity adjustments, if required, are manually entered from the keyboard.

6.3 Interface to a Personal Computer

ISIS 250 is essentially a real-time system, designed to give rapid identification of and response to alarm conditions. With increasing frequency, requirements are encountered for long term trending and analysis, interfacing with proprietary ship management programs, ship stability programs, etc.

These types of requirements are handled flexibly and economically by collecting selected ISIS 250 channel information and transferring this data via the ISIS 250 PC Data Collection Program to prepared files in a personal computer.

A special interface processor (DCU) is connected to an IBM, or compatible, personal computer by an RS232C link. The program provides a menu of displays in the PC which allow the user to:

- Set up the channels from which he requires to collect data.
- Determine the collection time intervals for each channel.
- Initialise the interface processor.
- Permit transfer of data to the P.C. with limitless manipulation possibilities.

ENGINEERING MARINE SYSTEMS USING ADA PROGRAMMING

by Charles J. Robert
and Terry Babin
TANO Marine Systems, Inc.

1. ABSTRACT

As marine control systems have evolved from pneumatic controls to primarily software-based monitoring and control systems, the need for a standardized software language has surfaced. Ada has been chosen as that language. Designing large, complex systems is facilitated by Ada's features. The support for "packages" allows designers to clearly define system components and their interfaces. For real-time systems, Ada's support for "tasks" provides for modular processing of asynchronous events.

Ada requires the programmer to stay within the strict guidelines of Ada syntax, making it easier for other programmers to adapt to already designed code.

Ada Compilers have been developed for a number of different machines ranging from IBM PC compatibles to 68000 VME computers. With minimal effort, systems designed for one environment may be adapted to another.

As we approach the 21st century, packages of Ada software will become readily available for purchase. When interfaced with a specific database and with a few polishing touches from the cognizant systems engineer, these packages will comprise complex marine systems. These modular systems will reduce the engineering effort considerably and allow the design effort to focus more on using more complex software calculations. These calculations will detect trends in machinery deficiencies and alert the operators that preventive maintenance is needed. Also expected to evolve with the purchasable Ada packages are new Ada compilers, which will produce faster code and be more efficient in regards to storage requirements.

This paper will describe the "lessons learned" in developing two marine control systems using Ada and will hope to set a trend for future Ada development. Current topics in software, processing, and engineering will be explained.

2. INTRODUCTION

Creating large, reliable, easy to maintain software systems is a complex, time-consuming task. As in any discipline, without proper tools and sound methodologies, the process of problem-solving is made even more difficult. "Programming languages are neither the cause of nor the solution to software problems, but because of the central role they play in all software activity, they can either aggravate existing problems or simplify their solutions" (1).

As applications for embedded computer systems become more sophisticated, the size and cost of programs will increase. Prior to Ada, no language existed which could allow a designer to deal effectively with their corresponding problem spaces.

After discussing Ada's history and dispelling a few myths that have cropped up about the language, this paper will provide suggestions on designing building blocks for a small-scale marine system.

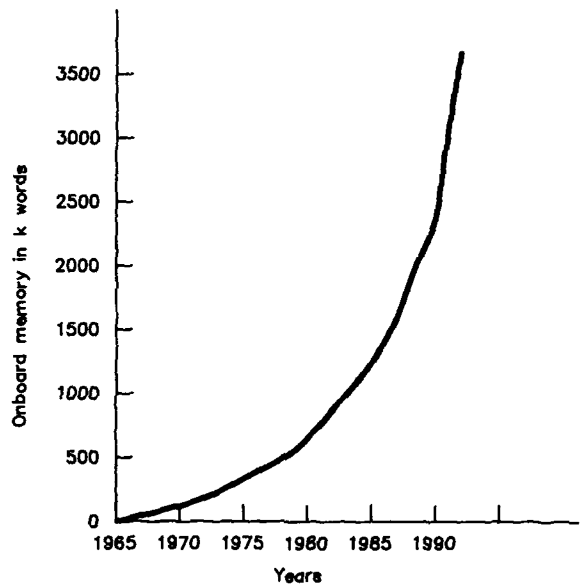
3. HISTORY OF ADA

Ada evolved over a period of time as a direct result of an effort to address what has become known as the software crisis. The following discussion summarizes some of the highlights of the process.

3.1 Early 1970's

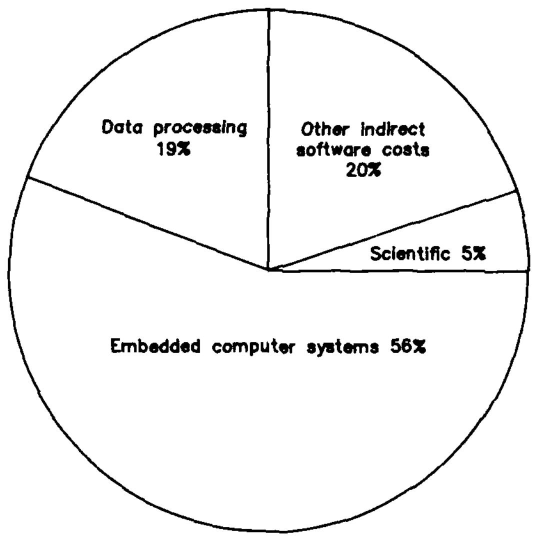
The United States Department of Defense (DoD) became concerned about the trend of rising costs of software for their computer systems. From 1968 to 1973, they noticed that there was a 51% increase in the cost of the systems although hardware was rapidly becoming less expensive. For applications such as weapons systems, cost was not the only issue; safety, reliability, and maintainability were also major concerns. The amount of code required for each successive system was growing exponentially. Figure 1 illustrates this growth.

Contributing to the software crisis was the usage of a diversity of languages which were ill-suited to embedded computer applications, causing support costs for these systems to skyrocket. "Some of these languages were so esoteric that only a few programmers were skilled in them, guaranteeing maintenance problems in years to come". (2) In 1973, software costs for embedded computer systems accounted for 56% of all their software expenses (see Figure 2), so the DoD turned its attention to that area.



Exponential growth in program size

Figure 1



Estimated DoD software costs in 1973

Figure 2

Due to the real-time nature of embedded computer applications, code written for the applications needed to be efficient. Assembly languages were normally considered for systems where efficiency and responsiveness to external events were critical. By this time, however, the computer industry began to understand the benefits of using high-order languages as far as software quality and system life-cycle cost reduction were concerned. Fortunately, hardware was becoming faster and compiler technology had improved to the point that such languages were being considered for real-time applications.

Unfortunately, although there were at least 450 languages in use for DoD systems, no one of them was very suitable for embedded applications.

3.2 1975

In January of this year, the High Order Language Working Group (HOLWG) was established. It included representatives from each of the armed services, a variety of other DoD agencies, and from the United Kingdom, West Germany, and France. Their purpose was to identify the requirements of high-order languages for use in DoD systems, evaluate existing languages, and recommend the exclusive use of a minimal set of languages.

In April, HOLWG distributed the requirements document known as STRAWMAN. Based on responses from military departments, other federal agencies, industry, and the academic community (in the United States and Europe), another document called WOODENMAN was produced. After further revision based on additional responses, TINMAN was created. This represented the desired characteristics for a high-order language.

3.3 1976

HOLWG, along with several contractors and individuals, began to evaluate existing languages based on the requirements set forth by TINMAN (which later became IRONMAN after consolidating additional reviewer comments). Also during this time, the DoD showed interest in reducing the number of languages in use by issuing a directive which required the use of an approved high-order language in defense systems (unless a different language could be proven more cost-effective), and which established a single point of control for each approved language.

3.4 1977

The process of evaluating IRONMAN was completed in January 1977 and led to the following conclusions:

-- A single language was desired.

-- No existing language was suitable for use as a common high-order language for DoD embedded computer systems.

-- A new language to meet the requirements was feasible.

-- The new language should be developed from an appropriate base. (3).

Under the direction of the Management Steering Committee, HOLWG began to oversee the development of a common high-order language (given the name DoD-1). HOLWG gave the responsibility for the language design contract to the Defense Advance Research Projects Agency (DARPA).

Since the DoD wanted the new language to be a common standard, they wanted its design to be of high quality. Also, they wanted it to embody a consistent philosophy so that it would be well accepted even outside the defense community. For these reasons, an international design competition was considered. Teams from around the world were to be solicited for designs which would then be evaluated. A few would be chosen to complete detailed designs for final evaluation. In April of 1977, a Request For Proposal (RFP) was issued internationally. The design effort took place in three phases.

In July, DARPA initiated Phase I of the design effort by selecting 4 of the seventeen responses to the RFP to continue in a six-month development period which started in August. The 4 participants were: SofTech, SRI International, Intermetrics, and Cii-Honeywell Bull. To prevent any bias on the reviewers' part, the identity of the participants was not made known. Instead, the proposals were color-coded Blue, Yellow, Red, and Green, respectively.

3.5 1978

In February of 1978, Phase I concluded. Each design was evaluated worldwide by almost 400 volunteers in 80 review teams. In March, as a result of this effort, two of the proposals (Red and Green) were selected for further refinement; thus began Phase II.

In parallel to the Phase II effort, HOLWG circulated a document known as SANDMAN which addressed issues concerning programming environments. Revisions to SANDMAN led to the PEBBLEMAN which was made public. Concurrent to this, HOLWG released STEELMAN, the final version of the language requirements document which eliminated the deficiencies uncovered during Phase I evaluation.

3.6 1979

Phase II was completed in March of 1979 and the two designs were considered through April. In May, HOLWG announced that the Green language (submitted by Cii-Honeybull Bull from France) had won the competition and that Ada would be the official name for the new language. The name was chosen in honor of Augusta Ada Byron, Countess of Lovelace, and daughter of the poet Lord Byron. "Ada Lovelace (1815-1851) was a mathematician who worked with Charles Babbage on his difference and analytic engines. She is noted for her early observations on the potential power of the computer. In particular, Ada suggested how Babbage's machines might be programmed much like the Jacquard loom, and for her work she is considered the world's first programmer." (4)

The announcement of the winner led to Phase III of the language effort, starting or commencing the test and evaluation period.

Volunteers were asked to implement an existing application in Ada. They were provided an Ada test translator and were allowed to participate in training classes which were established at the Navy Postgraduate School, the Air Force Academy, West Point, Georgia Institute of Technology, and the National Physical Laboratory (in England). During this time, a preliminary language reference manual was circulated so that selected experts could detect any flaws in the language design and correct any ambiguities in the manual. In a move to prevent dialects from forming, HOLWG initiated a contract to develop a compiler validation facility during this phase. Also, in the fall of 1979, an Ada Board, consisting of a group of distinguished reviewers was set up to manage any proposed language changes. After a public test and evaluation conference held in Boston in October 1979, over 500 reports from 15 countries were submitted. Generally, they concluded that the language design was acceptable but needed some slight modifications.

3.7 1980

HOLWG released a further refined document for the Ada programming environment known as STONEMAN. The final version of STONEMAN, distributed in February of 1980, became the basis for Ada Programming Support Environments (APSEs). In July, based on the reports generated in Phase III, the Ada reference manual was finalized.

In December, HOLWG was dissolved and the Ada Joint Program Office (AJPO) was created. Also, MIL-STD-1815 was established as the DoD standard for Ada.

3.8 1981

To continue the policy set forth by HOLWG of preventing dialects, AJPO applied for Ada as a trademark of the DoD in January 1981. "To be legally called an Ada compiler, a compiler must pass a suite of more that 3000 validation tests and be reevaluated annually". (5) Policy statements were obtained from each of the military services which indicated commitments to Ada as a standard, with a planned phaseout of all other languages for new embedded systems by the mid-1980's.

3.9 1983

On February 17, 1983, the Ada language reference manual was approved as an American National Standards Institute (ANSI) standard so that Ada could move out of the exclusive domain of the defense community and into the general computing industry. Ada's military designation was changed to MIL-STD-1815A.

4. POPULAR MYTHS ABOUT ADA

The earliest attempts at using Ada for real-time applications were not very successful due to immature compilers and because it was not clear how Ada should be employed for such systems. As a result, some myths have evolved.

One myth is that Ada was designed by a committee and therefore had inherent shortcomings. As can be seen in the previous discussion of Ada's history, Ada evolved via a design competition the results of which were subjected to review by thousands of computer science experts around the world. Although the potential complexity and confusion from such wide participation could have resulted in a confused product, it did not. Excellent leadership was provided by a select few professionals, and final decisions were made by one man (Dr. Jean Ichbiah) or by someone he selected. As a result, Ada has become a powerful and consistent tool for creating and managing large, complex software systems.

Another myth is that Ada is too large to learn. Actually, there are only 63 reserved words in the language. What follows are lists of these words grouped by category:

Program Unit and Block related words:
begin
body
declare
end
function
generic

package
procedure
return

Control flow related words:
case
do
else
elsif
exception
exit
for
goto
if
loop
raise
then
when
while

Words relating to task objects:
abort
accept
delay
entry
select
task
terminate

Words relating to other objects:
access
all
array
at
constant
delta
digits
is
limited
new
null
of
others
out
private
range
record
renames
reverse
subtype
type

Arithmetic and Boolean operators:
abs
and
in
mod
not
or
rem
xor

Other words:
pragma
separate
use
with

Anyone familiar with other high-order languages (such as Pascal) will have no trouble with about 40 of the above words (which are either exactly the same or are simply different words used in similar ways). Ada's power is partly due to the flexible use of some of these words; proficient use of the language does require a fair amount of study, but that is not a drawback.

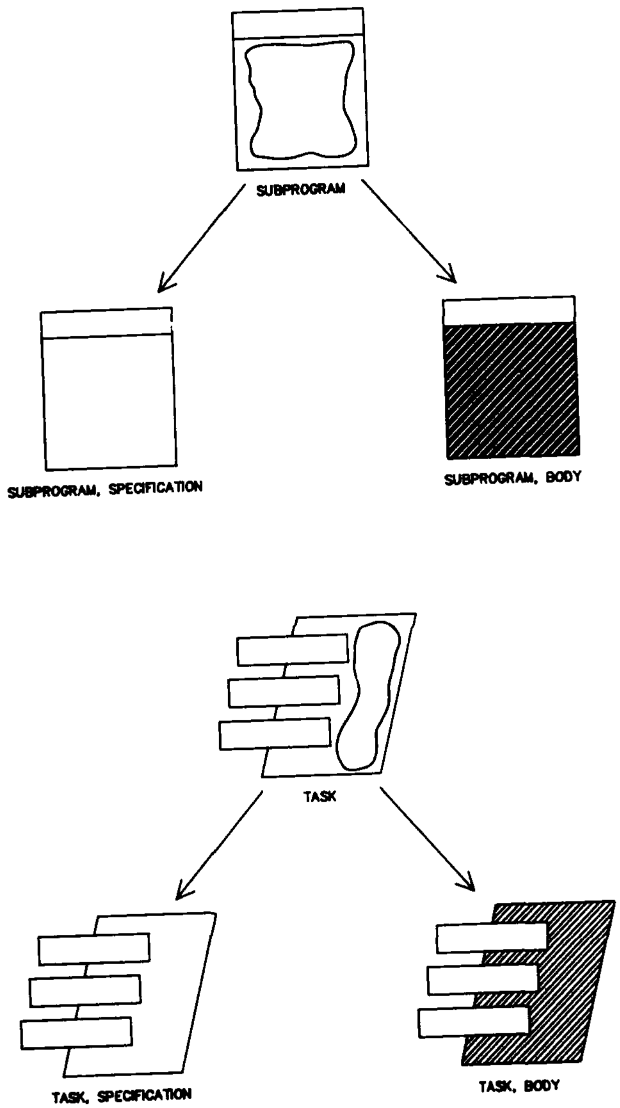
Also, Ada does incorporate a few unfamiliar concepts (such as packaging and tasking) which require a designer to make mental adjustments, but as with any development tool, in order to use novel features, the features must be learned.

Yet another myth actually has a factual basis; that Ada is too slow for real-time applications. Early compilers were immature but it is acknowledged that performance problems have largely disappeared over the last couple of years. Among the improvements are compiler optimization schemes, faster run-time executives, and alternative interrupt mechanisms.

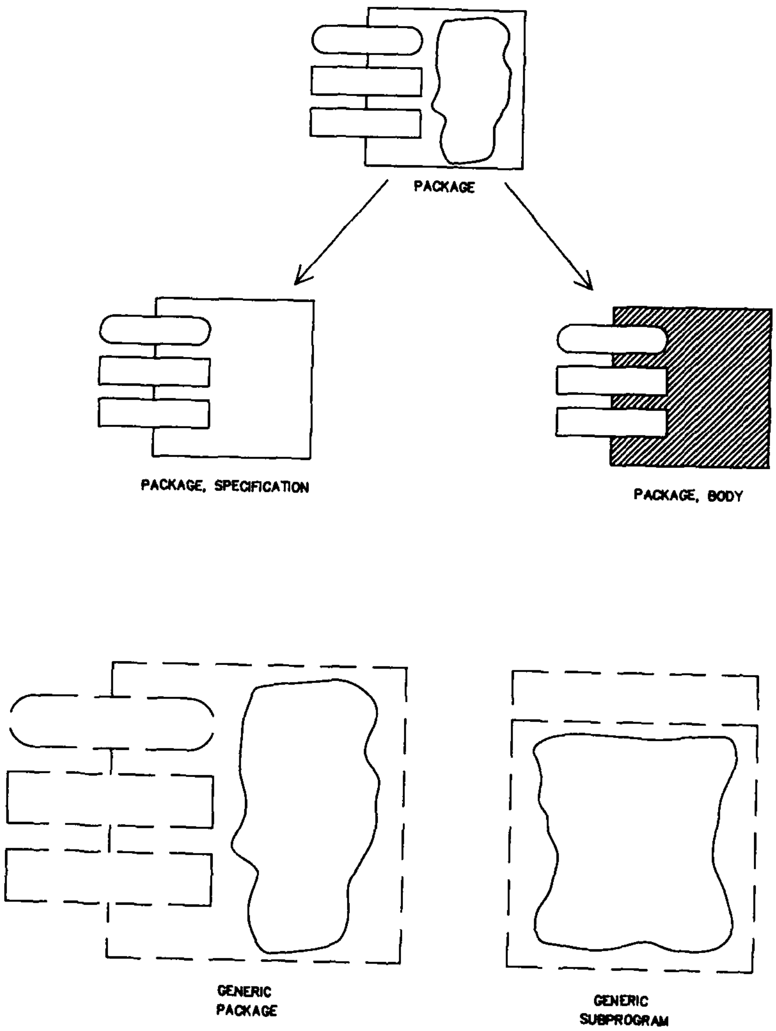
5. ADA'S FEATURES FOR MARINE SYSTEM COMPONENTS

In this section, a design for a simple small-scale marine system will be discussed. Included in the discussion are descriptions of Ada features that are used along with examples of applied software engineering principles. The objects in Figure 3 (a and b) are taken from the text **Software Engineering with Ada** by Grady Booch to pictorially represent Ada program units. These objects will be used throughout the remainder of this section to illustrate the components of the marine system. A brief description of each object follows.

A subprogram specifies a sequence of actions. It is either a **procedure** or a **function**. A subprogram is written as a



A Subprogram and a Task
Figure 3(a)



A Package and some Generic Units
Figure 3(b)

subprogram declaration - which specifies its name, formal parameters, and the result if it is a function - and a subprogram body.

A **task**, which operates in parallel with other parts of the program, is written as a task specification (name of task, name of entries, formal parameters of entries) and a task body, which defines its execution.

A **package** forms a collection of logically related entities or computational resources. It has two parts, a specification (which identifies the "visible" parts of the package) and a body. The visible parts of a package are those entities (objects, types, subprograms, and even other packages) which a user of the package can use directly.

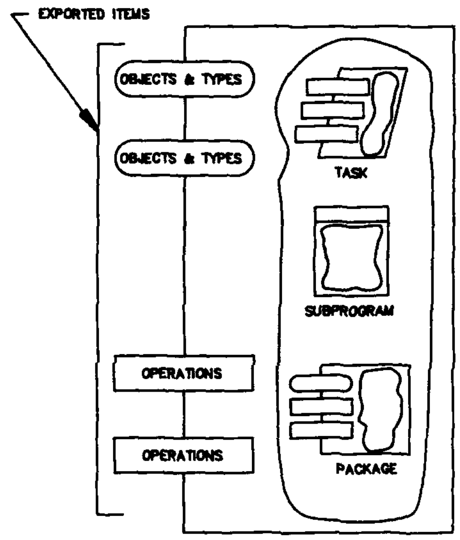
The curved arrows in subsequent examples using the components described above do not represent data flow but rather relationships between the components. For example, if a curved arrow were drawn from a subprogram to a package, it is said that the subprogram uses or "imports" at least one of the elements (types, objects, or computational resources) listed in the package's specification.

The body contains all resources which define the implementation of the package (which a user can not see since the details are not important at the user level). Figure 4 shows a package in more detail.

Visible entities are said to be exported. Note that a package body may contain any number and combination of other program units. If a program unit is to be exported its specification must appear in the package specification and only its body must be in the package body. (Actually, the body may not necessarily be physically contained within the package body; the Ada programming environment allows for separate compilation of such units.)

A **generic** is a parameterized module that serves as a template for other modules; it is sort of a high-level language macro. An engineer may define generic packages, procedures, and functions. "In a sense, generic units are to subprograms and packages as types are to objects." (6) Generics are useful since they need only be written and tested once.

One application for generics is to create abstract data structures such as queues (another is given in the discussion of a database), structures required in most real-time embedded computer systems.



Package

Figure 4

5.1 Database

At the heart of most systems is some central storage of data along with means by which the data can be stored and retrieved. These elements comprise the database.

Essential to a database is a collection of declarations which describe the records (and the tables that contain them) that comprise the database. This collection, in Ada, would normally be contained in a program unit known as a package. Below is a sample pseudo-code which could be used to describe a database:

```
package Database is

  type Table_Typ is (...);

  type Record_Location is
    record
      Table           : Table_Typ;
      Number_In_Table : Natural;
    end record;

  type Record_Typ1 is
    record
      .
      .
      .
    end record;

  type Record_Typ2 is
    record
      .
      .
      .
    end record;

  ...

  type Record_TypN is
    record
      .
      .
      .
    end record;

end Database;
```

Consider a routine to "Get" a record which accepts as input a **Record_Location** and which passes a complete record as output. Ada requires that there be a separate "Get" routine for each record type since, in general, each type would be different from another. It would be quite a burden for an engineer to code routines to "Get" (as well as "Put") data for each record since the routines would be identical except for the type of record being handled. The generic feature in Ada ameliorates this burden.

A generic package can be developed to define the routines needed to access records from the database. Once this is done, an engineer need only "instantiate" (that is, create an instance of) the package for each type of record. Below is a code sample for a package called **Record_Access**:

```
with Database; use Database;
generic
  type Record_Typ is private;
package Record_Access is

  procedure Get (Rec          :    out Record_Typ;
                 At_Location  : in     Record_Location);

  procedure Put (Rec          : in     Record_Typ;
                 To_Location  : in     Record_Location);

end Record_Access;

package body Record_Access is

  procedure Get (Rec          :    out Record_Typ;
                 At_Location  : in     Record_Location) is
  begin
     .
     .
     .
  end Get;

  procedure Put (Rec          : in     Record_Typ;
                 To_Location  : in     Record_Location) is
  begin
     .
     .
     .
  end Put;

end Record_Access;
```

The instantiations of the above generic package may actually be performed for each record type in the specification of another package so that all possible "Get" and "Put" routines can be made available to all users of the database. This would prevent other parts of the system which need access to the database from having to trouble with performing individual instantiations. The package **Database_Access** below does just that:

```
with Database;      use Database;
with Record_Access;
package Database_Access is

  package Record_Typ1_Access is new Record_Access (Record_Typ =>
Record_Typ1);

  package Record_Typ2_Access is new Record_Access (Record_Typ =>
Record_Typ2);

  ...

  package Record_TypN_Access is new Record_Access (Record_Typ =>
Record_TypN);

  procedure Get (Rec          :    out Record_Typ1;
                 At_Location  : in     Record_Location)
  renames Record_Typ1_Access.Get;

  procedure Put (Rec          : in     Record_Typ1;
                 To_Location  : in     Record_Location)
  renames Record_Typ1_Access.Put;

  procedure Get (Rec          :    out Record_Typ2;
                 At_Location  : in     Record_Location)
  renames Record_Typ2_Access.Get;

  procedure Put (Rec          : in     Record_Typ2;
                 To_Location  : in     Record_Location)
  renames Record_Typ2_Access.Put;

  ...

  procedure Get (Rec          :    out Record_TypN;
                 At_Location  : in     Record_Location)
  renames Record_TypN_Access.Get;

  procedure Put (Rec          : in     Record_TypN;
                 To_Location  : in     Record_Location)
  renames Record_TypN_Access.Put;
```

```
  procedure Get (Rec_Location :    out Record_Location;
                 Using_Key    : in     String;
                 From_Table   : in     Table_Typ);

end Database_Access;


package body Database_Access is

  procedure Get (Rec_Location : Record_Location;
                 Using_Key    : String;
                 From_Table   : Table_Typ) is
  begin
     .
     .
     .
  end Get;

end Database_Access;
```

These components, and the relationships between them, are illustrated in Figure 5.

The last declared procedure Get is added for completeness. The assumption is that the tables are indexed in some way and that some string (unique for each record in a given table) can be used to look up the location of the corresponding record.

5.2 Data Acquisition

Every marine system has a means by which data is acquired from ship machinery sensors. Some systems may have master computers which collect data via a serial line interface to a slave computer dedicated to data acquisition, others systems may have computers acquiring data from a sensor interface on a local bus. In any case, these details are not generally essential to the top-level design of a system.

Two of the fundamental principles for managing software complexity are abstraction and information hiding. "The essence of abstraction is to extract essential properties while omitting inessential details" (7). Whereas abstractions extract the essential details of a given level, "the purpose of hiding is to make inaccessible certain details that should not affect other parts of the system" (8).

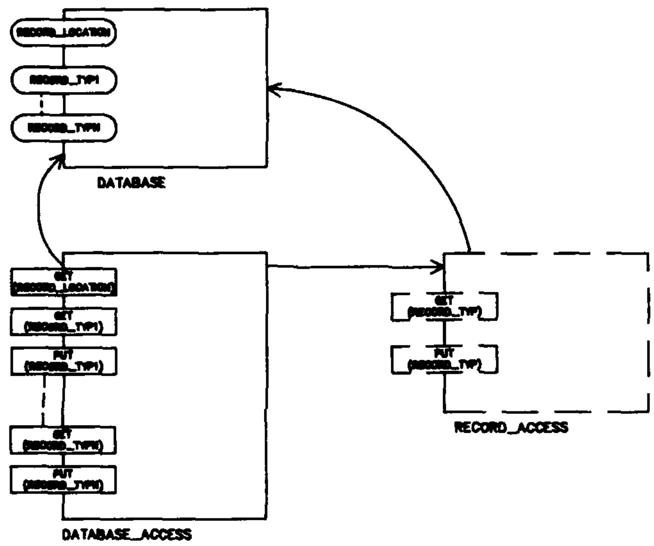


Figure 5

In his text, **Software Engineering with Ada**, Booch exemplifies these principles with a discussion of a disk drive. Software may interact with the device at several levels:

- as a collection of logical files
- as a mass storage organized by tracks and sectors
- as a collection of addressable bits of storage
- as a physical device requiring control and data signals

As can be seen by this example, by forming a ladder of abstraction, an engineer reduces the number of entities with which he needs to be concerned at any given level. Also, note how certain information about the device gets hidden as the ladder rises. A user viewing a disk as a collection of logical files would not be permitted to access, for instance, a given sector.

A typical device in a marine system is an Analog-to-Digital converter. For the following discussion, assume that a single converter is capable of digitizing any one of many multiplexed analog channels at any one of several gains. At the lowest level of detail, there is a device (the converter) which must be programmed through storing information in control registers and which stores results in data registers.

A user interested in using the converter should not have to be concerned with details of how to use the control registers nor with how the results are stored, but rather with what operations are available. Here is a first step in the ladder of abstraction:

```
with System;         use System;
package A_to_D_Converter_Drivers is

  type     Gain_Value       is (...);

  subtype Count_Value      is Integer range ...;

  subtype Mux_Address      is Integer range ...;

  procedure Get (Counts     :    out Count_Value;
                 At_Address : in     Address;
                 On_Mux     : in     Mux_Address;
                 With_Gain  : in     Gain_Value);
```

```
  procedure Start_A_to_D_Conversion
          (At_Address    : in     Address;
           On_Mux        : in     Mux_Address;
           With_Gain     : in     Gain_Value);

  function A_to_D_Conversion_is_Done
          (At_Address   : in     Address) return Boolean;

  procedure Get (Counts     :    out Count_Value;
                 At_Address : in     Address);

  Timeout : exception;

end A_to_D_Converter_Drivers;
```

The above package specification provides a tool box of routines for using the A to D converter. Consider the procedure **Get** which is declared immediately after the types and subtypes. Its implementation will actually use the procedure **Start_A_to_D_Conversion**, the function **A_to_D_Conversion_is_Done**, and the last declared procedure **Get** to achieve the intended result. Having the latter three subprograms appear in the package specification may seem to be a violation of the "information hiding" principle, but consider this: a result from a digitizing request is not obtained instantaneously. In the case of the first **Get** procedure, control will not return to the caller until the converter is finished (or until the **Timeout** exception is raised). If the caller has strict time constraints and wishes to perform some useful processing (such as converting the previous digitized value to engineering units) while a converter is busy then it has the facilities to do so.

The user of the package **A_to_D_Converter_Drivers** is typically called the poll task. Consider the very simple package below:

```
package Poller is

  task Poll_Task is
    entry Start;
  end task;

end Poller;

with Database;                  use Database;
with Database_Access;           use Database_Access;
with A_to_D_Converter_Drivers;  use A_to_D_Converter_Drivers;
package body Poller is
```

```
  task body Poll_Task is
  begin
    accept Start;
    .
    .
    .
  end Poll_Task;

end Poller;
```

All that is left to do now to form a (very bare) system is to define a procedure to represent a starting point for the main program. This could be done as follows:

```
with Poller; use Poller;
procedure Main is
begin
  Poll_Task.Start;
end Main;
```

The diagram for the complete system thus far is given in Figure 6.

Note that there are no context clauses (that is, "with" statements) in the specification. **Poller** does not require the services of any other packages and only exports one object, that is, the poll task (which has one entry point). The specification can appear in the top-level of design in any system requiring such an operation; it is not tied to anything specific.

The body of **Poller** is different in this respect, however. The implementation is not important to the top-level design so is not visible.

The code in the body may read data from a device on the local computer bus or interface to a serial device to communicate with a remote data acquisition computer. Also, **Poll_Task** may not actually do any polling at all. It may simply start up other tasks (perhaps in separate packages), each of which perform data acquisition (perhaps to different types of devices). Clearly, there is a need for modularizing the polling software especially for complex systems involving a variety of devices.

Modularity is another fundamental principle for managing the complexity of large software systems. It applies to the physical architecture of the system. Consider the example above where the

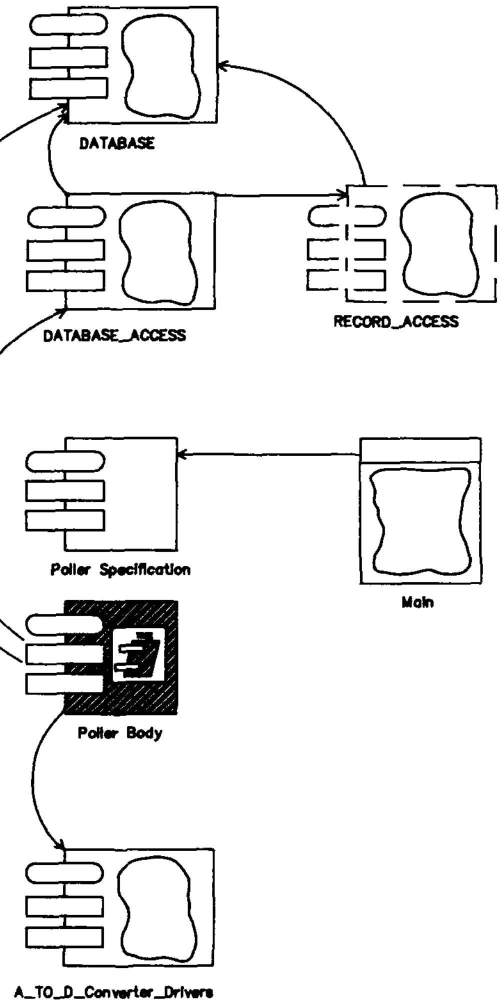

DATABASE
DATABASE_ACCESS
RECORD_ACCESS
Poller Specification
Main
Poller Body
A_TO_D_Converter_Drivers

Figure 6

Poller body may contain two packages which contain computational resources to poll dissimilar devices. Here, the top-level module **Poller** has been decomposed into two lower-level ones. This example also exhibits still another principle of software engineering, namely, localization. The two lower-level packages are logically related, in a sense, so are collected in the same physical module. Localization also implies that modules are as independent as possible. Certainly, that is the case here since neither package is dependent on the other.

The diagram in Figure 7 illustrates the above example.

The previous discussion exemplified how one may begin to define the building blocks for a marine system. What follows in Figure 8 is a fairly complete, high-level design for a small-scale system using the components already developed.

The poller in Figure 8 uses routines in **Analog_Monitors** to determine whether or not current values are outside normal operating parameters. If one is then **Analog_Monitors** uses routines in **Alarm_Processor** which interfaces to other packages which may operate console devices (such as lamps and horns) (**Annunciator_Driver**) and a printer (**Printer_Driver**) to annunciate the condition, or may take special action such as interfacing to some package to shut down machinery (**Control_Outputs**). Other features of the depicted system include a package which performs the storage and retrieval of historical information (**Historical_Data**) and an **Operator_Interface** package which supports all operations and displays relating to some CRT device. Note how naturally Ada components may be used to describe a high-level design.

6. *CONCLUSION*

Most programming languages can be used to solve any problem that arises in an embedded computer system. Differences between the languages, however, determine whether a solution can be expressed easily and naturally; this clearly affects the expense of development. Ada is designed specifically for real-time, large-scale, complex, long-lived applications. Its success is due largely to the unprecedented development and review effort undertaken by the United States Department of Defense. According to most sources, it has been and will continue to be an effective tool for those systems for which it was designed. Figure 9 illustrates the trend that software costs have taken over the past decade; clearly, the right tool is essential.

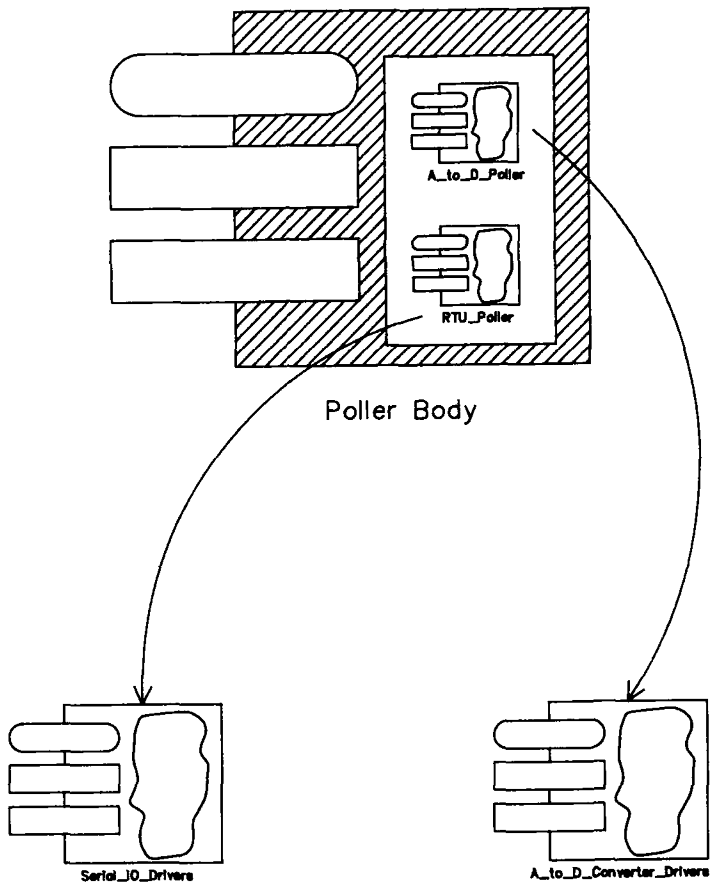
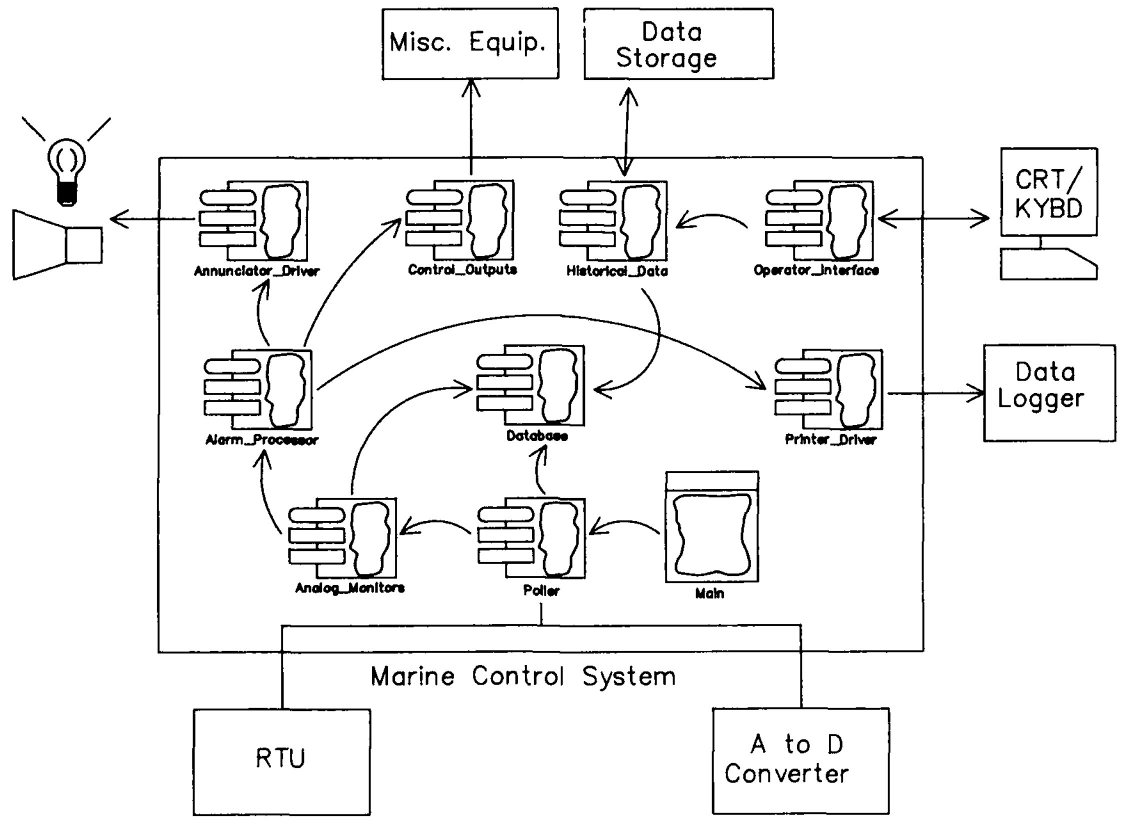
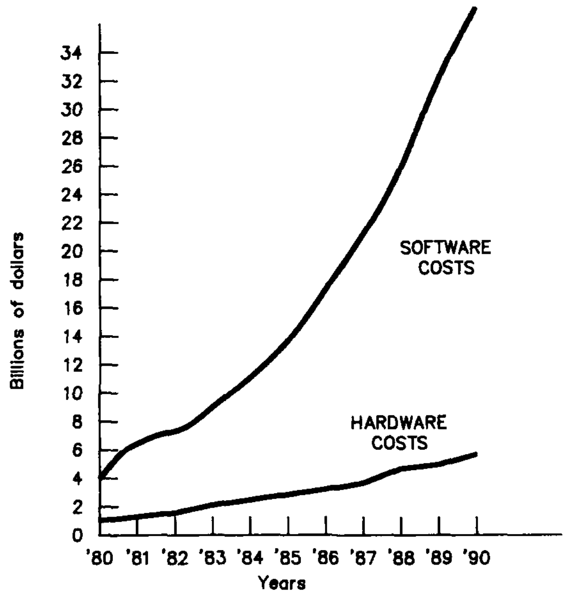

A_to_D_Poller
RTU_Poller
Poller Body
Serial_IO_Drivers
A_to_D_Converter_Drivers

Figure 7



Typical Marine Control System
Figure 8



DoD hardware/software cost ratio

Figure 9

The trend that the cost of software is taking is not unique to the DoD; the private sector is also affected. Developers can progress toward more economical systems by creating re-usable Ada software components. With the strict requirements that govern Ada compilers, and with the proper application of modern software engineering principles, designers can ensure that their components are highly portable.

7. REFERENCES

(1) D. A. Fisher, "A Common Programming Language for the Department of Defense - Background and Technical Requirements," Institute for Defense Analysis, Report P-1191 (June 1976): p. 8.

(2) Steven H. Leibson, "Real-time Ada", EDN, August 17, 1989, p. 104.

(3) W. A. Whitaker, The U. S. Department of Defense Common High Order Language Effort, SIGPLAN Notices, February 1978, pp.7-8.

(4) Grady Booch, Software Engineering With Ada, Second Edition, p. 21.

(5) Leibson, op. cit., p.105.

(6) Grady Booch, op. cit., p. 244.

(7) D. T. Ross, J. B. Goodenough, and C. A. Irvine, "Software Engineering: Process, Principles, and Goals," Computer (May 1975), p. 67.

(8) Ibid.

RAST MK III - THE CONTROL ASPECT OF A NEW GENERATION HELICOPTER HANDLING RECOVERY SYSTEM

by Lieutenant Commander N. Leak BEng MSc PEng
and Lieutenant(N) G.S. Brown BEng
Department of National Defence, Canada

1. ABSTRACT

The RAST MK III system represents the next generation of shipboard helicopter recovery and handling systems. RAST MK III is a fully integrated, software-based system. Other significant differences between RAST MK III and current RAST systems include the elimination of the hauldown cable and yaw restraint systems. The RAST MK III system features software written in Ada and designed to the Defense System Software Development Standard (DoD-STD-2167A), the application of real-time photogrammetry, enhanced pilot visual cues display, and a redesigned aircraft handling system.

This paper discusses the development effort of the RAST MK III R&D project with the Helicopter Position Sensing Equipment (HPSE) developed being emphasized throughout.

2. INTRODUCTION

In the early 1960s the Royal Canadian Navy (RCN) defined a requirement to conduct Anti-Submarine Warfare (ASW) operations using relatively large helicopters operating from small ships during virtually any condition of weather and visibility. The Navy needed a system which would enable it to safely deploy and recover its main weapon/sensor system, day or night, in conditions up to sea-state five (30 degrees of roll and 9 degrees of pitch) and in relative winds up to 50 knots [1].

The Helicopter Hauldown and Rapid Securing Device (HHRSD), or what was more commonly referred to as the "Beartrap", was specifically developed for the RCN. The HHRSD has become an essential and important part of Canadian and many foreign Navies ASW operations. The HHRSD system presently in use remains a very capable system, however, it is nevertheless a twenty-five year old design. The experience that has been gained over the years and the development of technology, over the same period, indicates that an improved system may be achievable.

The RAST MK III (Recovery, Assist, Secure and Traverse) Project is a three phase Research and Development effort to address some of the fundamental shortfalls of the current system with state-of-the-art technology.

This paper will briefly describe the development effort of the RAST MK III project. The Helicopter Position Sensing Equipment (HPSE) will be emphasized throughout. This was one of two main sub-systems (the other being Ship Motion Prediction - SMP) that was identified as requiring the most development work. Through further examination of the proposed operational scenario, it was clearly recognized that the HPSE, and not Ship Motion Prediction, was vital to the successful operation of the overall system. At the end of the development section there will be a brief discussion on the proposed system architecture and a system operational overview. Finally, a short description of the system software design methodology will be addressed.

3. BACKGROUND

3.1 Current System

The HHRSD provides a means of mechanically securing the helicopter after landing, and then straightening and traversing it using a minimum of flight deck personnel (Fig. 1). As an interesting aside, the original beartrap was a wireless system, calling for a "free-deck" landing. It was only during initial trials that the capture area of the trap was found to be too small for "free-deck" landings and that some form of recovery assistance would be required. The addition of the hauldown cable was not totally welcomed by the pilots, and it was some time before the idea of being tied to a wire gained general acceptance [1].

3.2 RAST MK III

The RAST MK III project originated in September 1984, with an unsolicited proposal from Indal Technologies Incorporated (ITI) of Mississauga, Ontario. The concept eliminates the hauldown wire and therefore the requirement for its associated below deck machinery. The work conducted, by Indal Technologies Incorporated, has consisted of a concept feasibility study, a preliminary conceptual design and the current phase, the production of an Advanced Development Model (ADM) for full scale, at sea, operational evaluation.

Figure 1 : "Flying the wire," necessary with the HHRSD system will become a thing of the past with the "wireless" RAST MK III. (DND photo)

4. CONCEPT FEASIBILITY STUDY

A study was conducted by ITI in 1986 to determine the conceptual feasibility of the RAST MK III proposal.

The first part, as with any feasibility study, was to identify the assumptions, define the required mathematical models and the data to describe the helicopter dynamics, ship motion, and airwake responses. These were defined as follows [2]:

a. the Sikorsky SH-60B Seahawk was selected as the helicopter model since it was the most representative of a generic NSA (New Shipboard Aircraft) air vehicle on which complete, unclassified, dynamic data was available;

b. the FF-1052 USN Knox class ship was selected to represent the CPF (Canadian Patrol Frigate) on the basis of being closest in size and having a complete set of unclassified ship motion and airwake turbulence data;

c. the definition of the worst case wind-over-deck and sea state operational envelopes were based on the current definition for the shipboard operation of the CH-124A Seaking helicopter;

d. the determination of clearance (i.e. Rapid Securing Device (RSD) trap size and location) and allowable hover dispersions were determined by examining the DDH280 Tribal class flight deck and Seaking airframe dimensions, since these represented the worst case conditions in Canadian naval operations;

e. the dynamic model was based on a linear six degrees of freedom stability and control model of the air vehicle and used a stochastic covariance approach to yield dispersion data directly. The model was developed around two commercially available software packages designed to run on an IBM PC. They were:

   (1) the transfer function program (TRFN) - the basic function of the transfer function program (TRFN) is to compute various transfer function elements from system equations. The program can evaluate the transfer function denominator or any specific numerator, coupling numerator, or coupling-coupling numerator. For the denominator or any numerator, the program computes and prints the roots of that polynomial in S, the highest order nonzero coefficient, and the lowest order nonzero coefficient. The program can also store specific transfer function on files that are compatible with

the second software package Program CC. The system equations are assumed to be in the matrix form:

$$A(s)X(s) = B(s)Y(s)$$

where A(s) is an N x N matrix of quadratic elements in s; N < 37

X(s) is an N x 1 vector of dependent variables

B(s) is an N x M matrix of quadratic elements in s; M < 19

Y(s) is an M x 1 vector of forcing functions (independent variables)

(2) Program CC (Complete Control) is a computer-aided control system design package. Some of the special features of Program CC are "user friendly" input, output, and symbolic manipulation of transfer functions; partial fraction expansion; interactive graphics with cursors; frequency and time domain LQG (Linear Quadratic Gaussian) control method; state space algebra; and macro capability.

The second phase of the concept feasibility study determined, using the dynamic models described above, that the preliminary RAST MK III concept was both feasible and realistic. ITI produced a preliminary design of the RAST MK III system with the following major system components [3]:

a. some form of a Rapid Securing Device (RSD) to provide the securing function together with some means of straightening the helicopter and traversing it into the hanger;

b. some means of providing the necessary positional data for display and/or control purposes for measuring the helicopter position. Both relative and absolute position would be required since the helicopter is to be guided to follow the mean ship's course and not "chase" the ship's roll and pitch. This would imply some sort of relative position measuring system and a ship motion sensing and prediction package; and

c. some means of providing landing data to the pilot.

5. DESIGN CONSIDERATIONS

In developing RAST MK III, the major design considerations centered around the known deficiencies and disadvantages of the current system. The essential difference between the new concept and the current HHRSD and RAST MK I systems would be the elimination of the hauldown cable and tail guide winches. Specifically, RAST MK III had to [1]:

a. provide for an integrated secure-and-traverse system;

b. allow day-and-night helicopter operations in up to sea-state five conditions;

c. eliminate the requirement for the recovery assist cable and tailguide winches;

d. eliminate the requirement for any personnel to be on deck during hover, landing, recovery or traverse stages;

e. decrease the existing time required for landing, straightening and traversing;

f. eliminate the requirement for below-deck equipment, and decrease system weight and space;

g. reduce system complexity, thereby reducing life-cycle cost and ILS requirements and improving reliability and maintainability;

h. be compatible with all RAST configured naval helicopters with minimal aircraft-fitted equipment, and be applicable to all sizes of naval ships; and

j. operate with surface or flush mounted tracks on the ship and be compatible with current track installations.

The identification of the major components of the system revealed the areas requiring the most development work were the helicopter position sensing and the ship motion prediction components. The approach and landing scenario for a helicopter using the RAST MK III system will be very similar to that currently used for "free-deck" landings with the HHRSD system. Consequently, by examining this scenario it was recognized that the ship motion prediction component, while important, was not a critical element in the landing process. The RAST MK III concept requires that the helicopter position be known at all times. This information needs to be passed to the pilot and to the controller of the RSD, in order to track the helicopter movement fore and aft in relationship to the ship. Therefore, the helicopter position sensing component becomes the "master controller" of the RAST MK III operation.

6. HELICOPTER POSITION SENSING OPTIONS

As already stated, real time helicopter position information is essential for the operation of the overall system. A survey of current technologies which were potentially capable of meeting the requirements defined in the feasibility study was conducted. A summary of ITI's report "Recovery Assist, Secure and Traverse System (RAST) Mark III - Helicopter Position Sensing Options" is detailed below [4]. All the position sensing proposals received as a result of the survey conducted by ITI, can be classified as either:

a. a multiple range measurement system for which the distance to the helicopter is measured; or

b. a multiple angle measurement systems for which the angle to the helicopter is calculated.

6.1 Multiple Range Measurement Systems

Multiple range measurement systems were found to have the potential for very compact, rugged and economical sensors, since they are fundamentally "simple". The sensors are not required to image the target, but rather just transmit and/or receive data. However, one disadvantage of this type of system is the limited range resolution, but this deficiency is being improved in the rapidly developing field of laser rangerfinders.

a. Acoustic Time Of Flight. It was recognized early in the program that acoustic time in flight measurement could be an inexpensive solution provided problems associated with air turbulence and background noise could be overcome. The theory of operation has a free running transmitter of an ultrasonic pulse train mounted on the helicopter close to the probe. The transmitter operates at a frequency of 40-50 kHz with a periodicity of 20 Hz.

A series of tests were conducted using Bell 205 and 206B helicopters at the National Aeronautics Establishment, Ottawa, by Canadian Astronautics Ltd under contract by ITI. The goal of this work was to obtain experimental evidence as to the feasibility of this solution and to derive a preliminary estimate of the positioning performance with helicopters in a low hover. However, difficulty with this concept was experienced in that the speed of sound is strongly dependent on the air temperature implying that some means of determining the air temperature very accurately would be required. Also, the measurement accuracy was severely limited by steady state wind-over-deck, airwake turbulence, vortices produced by the helicopter rotor downwash and the background noise due to the engines and rotor. From the results of this work it was determined that the acoustic based concepts pose a higher

development risk than competing optical concepts and did not appear to provide a more economical approach due to the added developmental costs.

b. Laser Rangefinders. Laser rangefinders measure the round trip travel time of laser light to return to the sensor from the target. The recent developments of high pulse power diode lasers indicate that laser rangefinders could be developed into compact, rugged and reasonably economical sensors. It has been found that lasers can operate over the wide range of weather conditions required for helicopter operations at sea. One of the greatest risks associated with the use of conventional lasers is the intensity of the light. Flared lasers may reduce the light intensity in terms of eye safety; however, there remains the problem of providing a stabilized platform and tracking system. This solution remains dependent upon sub-nanosecond resolution of pulse timing for short range measurements. This resolution is deemed to be only achievable in the laboratory and not yet suitable for field application and the use of flared lasers therefore, was not pursed.

c. Electromagnetic Phase Measurement. Electromagnetic phase measurement measures the phase of the modulation of either a radio frequency electromagnetic carrier (15 MHz modulation, 150MHz carrier) or an infrared (IR) laser diode signal. Phase measurement systems have been successfully demonstrated in tightly controlled environments but are seriously degraded by multiple propagation paths (i.e. reflections from surrounding material) such as are expected with the helicopter landing system. There were no such products or experimental hardware available on the market, thus this solution was rejected due to the developmental risk associated with it.

6.2 Multiple Angle Measurement Systems

In general, angular measurement sensors are more complicated than range measurement sensors and therefore tend to be less robust and inherently more expensive than range sensors; however, the technology is more mature. All of the multiple angle measurement system proposals received were based on operational or laboratory systems. It was therefore expected that the resulting development cost should be lower. Multiple angle measurement systems can be subdivided into optical imaging systems or mechanically scanning systems.

a. Optical Imaging Systems. Optical imaging systems work either in the visible or the near-infrared, and establish the angle to a target by measuring the image position of the target in an electronic camera. These systems have been successfully demonstrated in both laboratory and indoor industrial environments, with a number of commercial systems available. The major

shortcomings of optical imaging systems are their relatively narrow field of view, and inability to cop with direct sunlight. Depending on the type of focal plane device (electronic camera) used and whether single or multiple targets are used, there are several different variations of optical imaging systems as follows:

(1) One-Dimensional CCD Camera System. The one-dimensional CCD (Charged Coupled Device) system generally uses an infrared (IR) light emitting diode (LED) as the source on the moving body. It is this source which is imaged by two or more pairs of vertical and horizontal cameras. In order for this solution to be a viable option, the problems associated with direct sunlight needed to be resolved.

(2) Two-Dimensional Photogrammetry System. Two-dimensional photogrammetry is based on the NRCC (National Research Council of Canada) Real-Time Photogrammetry System (RPS) approach. The RPS is a position measurement system based on an electronic camera used to determine the position and attitude of an object from its image. The object is provided with a source array, such as IREDs or lasers, of known geometry and configuration. The camera, of known focal length and placed in a known location relative to a some reference point, is aimed such that the source array is in its field of view. The system then determines the position of the source elements of the array from the two-dimensional image formed in the camera. Photogrammetric and trigonometric techniques are then used to calculate the three-dimensional position and attitude of the object from the image formed in the camera [5]. The Canadair CL 227 UAV (Unmanned Air Vehicle) was successfully trialed at sea in August 1989 employing a position sensing system based on optronics. These photogrammetric and trigonometric algorithms used are the subject of Canadian and U.S. patents [5].

The RPS photogrammetry technology is the only known approach which can provide both position and attitude information (6 degrees of freedom) from a single camera. This approach has been successfully demonstrated in industry and with the Space Vision System (SVS) used to control the Shuttle Manipulation Arm ("Canadarm") built by Spar Aerospace of Montreal, Quebec.

(3) Two-Dimensional CCD Camera System. The two-dimensional CCD camera system is the same as the one-dimensional system except that if the helicopter mounted target beacon is mounted on or near the probe, there is no need for attitude information. A second camera can be used to provide three-dimensional data thus making the

photogrammetry approach overly complex for this solution.

b. Mechanically Scanned Systems. It has been found that high accuracy angular measurements can be made using mechanical systems which scan for an optical or microwave narrow beam width. Since a narrow beam width can be used, the signal to background ratio is high, and mechanically scanned systems can have very large fields of view. These systems are as follows:

(1) Laser Scanning System. The laser scanning system uses three rotating planes of laser light from scanners mounted on the ship's deck with a single photo-diode array mounted on the helicopter near the probe. The timing of the laser scans received by the helicopter equipment are telemetered to a shipboard computer to establish angles and/or processed into helicopter position co-ordinates. One of the significant advantages of this approach is that images are not formed therefore water, oil and a reasonable amount of solid debris can accumulate on the optical ports without affecting system performance. The major disadvantages of such a system include the need for more complex airborne equipment, the need for data telemetry between aircraft and the ship, the need for a rotating-mirror mechanisms, and the use of lasers which poses a potential problem of eye safety.

(2) Microwave Scanning System. A microwave scanning system using two microwave receivers on the flight deck together with an aircraft mounted transmitter was originally proposed for UAV/RPVs (Unmanned Air Vehicles/Remote Piloted Vehicles). The microwave scanning system uses a transmitter in the aircraft to transmit a modulated fan-shaped directional microwave beam towards the receivers mounted on the flight deck. The pulse repetition rate varies with the angle of transmission in a controlled fashion such that the mechanically scanned directional receivers indicate the angle from the aircraft to the receiver. From this data it is possible to calculate the aircraft position.

7. THE REQUIREMENTS

7.1 Initial Requirements

The initial requirements for the helicopter position sensing system were based on the assumption that the helicopter position had to be measured under the worst case high and low hover positions (corresponding to maximum ship's roll and pitch) and needed to be precise for use both in RSD tracking and for closed loop autopilot control of the aircraft. In order to comply with this requirement, multiple sensors (at least 4) would be required

around the edge of the flight deck which would result in an increase in system complexity and cost.

7.2 Factors Affecting Position Sensing Requirements

Several factors affecting the position sensing requirements came to light during the course of the Concept Feasibility Study which suggested a lesser measurement volume, a lower accuracy and only x and y measurements were required. These included:

a. a move away from closed loop autopilot control of the aircraft. This closed loop autopilot requirement would result in considerable expense and complexity, as well as requiring additional equipment on the already space critical aircraft. It was realized that the pilots would be reluctant to accept and use such a system in close proximity to the hanger face during a landing at sea;

b. a re-assessment of pilot landing cues and procedures. The current shipboard landing procedures and the pilot's requirement for positional cues to assist in landing were examined. From this study it was determined that the position sensing requirements could be prioritized as follows:

   (1) fore/aft position,

   (2) lateral position desirable, and

   (3) height unnecessary.

c. a reduction in the sensing accuracy. It was found that the accuracy for both high hover position and the measurement volume at low hover could be reduced. This was based on the fact that pilots descend to low hover only during low amplitude ship motions and then require position relative to the instantaneous deck position.

8. TECHNOLOGY TRADE-OFFS

The revised position sensing requirements favoured the use of an angle measurement system. The preferred option was the optical imaging system because the required data could be provided with a single sensor.

The one-dimensional linear CCD array system did not appear to offer any advantage over the more conventional two-dimensional array system. The only apparent advantage of the one-dimensional linear CCD array system was an increased processing speed and/or accuracy. However, this was associated with a higher production cost. Provided the target beacons are placed on or near the probe,

there is no requirement for the full photogrammetry approach, as only position (and not attitude) is required.

A similar two-dimensional CCD camera system could also be utilized to provide x and y data. This approach would still use multiple target beacons to help make recognition more reliable. Based on the facts available to date, the two-dimensional CCD camera system is the preferred option being pursued.

Should the sun image rejection problem prove impractical then the microwave landing system will be the fallback option.

9. POSITION SENSING SYSTEM DEVELOPMENTAL RESEARCH

The major concern with the use of any of the camera solutions was their inability to properly discrimination the target in direct sunlight due to camera-imaging blooming, and image distortion caused by water droplets or an oil film on the camera lens. In order to resolve these issues Indal Technologies Incorporated (ITI) under took a series of feasibility experiments [5, 6]. The results of these experiments are summarized below.

9.1 Camera Tests

A series of comparison tests were carried out with CCD and CID (Charge Injected Device) cameras to evaluate their performance. The CID 776 (CID Technologies Inc.) camera was found to be the most suitable camera for imaging IREDs in the presence of full sunlight. In this application the CID sensor technology is superior to the CCD technology due to non-blooming, non-streaking, and contiguous pixels characteristics. Since, the CID-776 has greater resolution than any of the other cameras and it can be qualified to military specifications, it has been incorporated into the design of the RAST MK III HPSE (Helicopter Position Sensing Equipment).

9.2 Position Sensing Sunlight Experiment

A series of sunlight experiments demonstrated that a better lens could reduce the internal reflections thus reducing optical blooming. These experiments also revealed that a brighter beacon using closely packed multiple IREDs increased the beacon-to-sun image ratio. The first experiments involving camera/lens/filter/beacon arrangements in bright sunlight were studied to determine the best image and least sky interference. Furthermore, it was determined that simple processing methods, such as thresholding and taking centroids in a "window", would not be able to cope with these images.

The second round of investigations involved experiments using advanced software processing methods for beacon location and measurement in the presence of sunlight. The capability of

advanced software processing methods to locate and measure the IRED beacon target was successfully demonstrated, however the results indicate that IREDs are marginal at best under static conditions, and fall far short under dynamic conditions. In order to overcome this problem the LED matrix would need to be increased. However, this would compound the thermal problems already associated with IRED use.

The laser diode option for beacons was then investigated. The laser diode has a higher power and efficiency, lower wavelength (higher camera sensitivity) and a narrower bandwidth (temperature stabilized). The laser diode approach solved the skin temperature and reliability problems associated with the use of IREDs. The proposed laser diode beacon system will consist of one 5W laser diode (800 nm) per helicopter side, feeding 4 beacon locations via fibre optic links. The beacons will be class 1 (basic lasers are class 4) which will render them <u>fully eye-safe</u> [6].

10. SYSTEM ARCHITECTURE

Having overcome the sunlight rejection problem, a configuration was then proposed for the advanced developmental model. ITI's proposal for the RAST MK III ADM system consists of the following major components [7]:

a. Rapid Securing Device (RSD);
b. Traverse Winch & Hydraulic Power Unit (TWHPU);
c. Helicopter Position Sensing Equipment (HPSE);
d. Pilot Cues Controller (PCC);
e. Ship Motion Prediction Equipment (SMPE);
f. Pilot Visual Cues Display (PVCD);
g. Operator Control Console (OCC); and
h. Local Control Unit (LCU).

System architecture diagrams shown in figures 2 and 3 graphically represent the described major components.

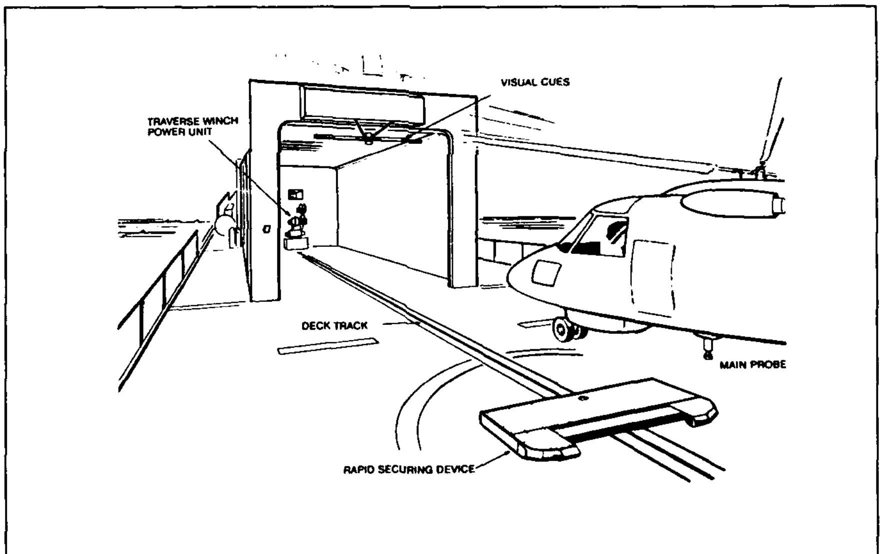


Figure 2 : RAST MK III System Elements [1]

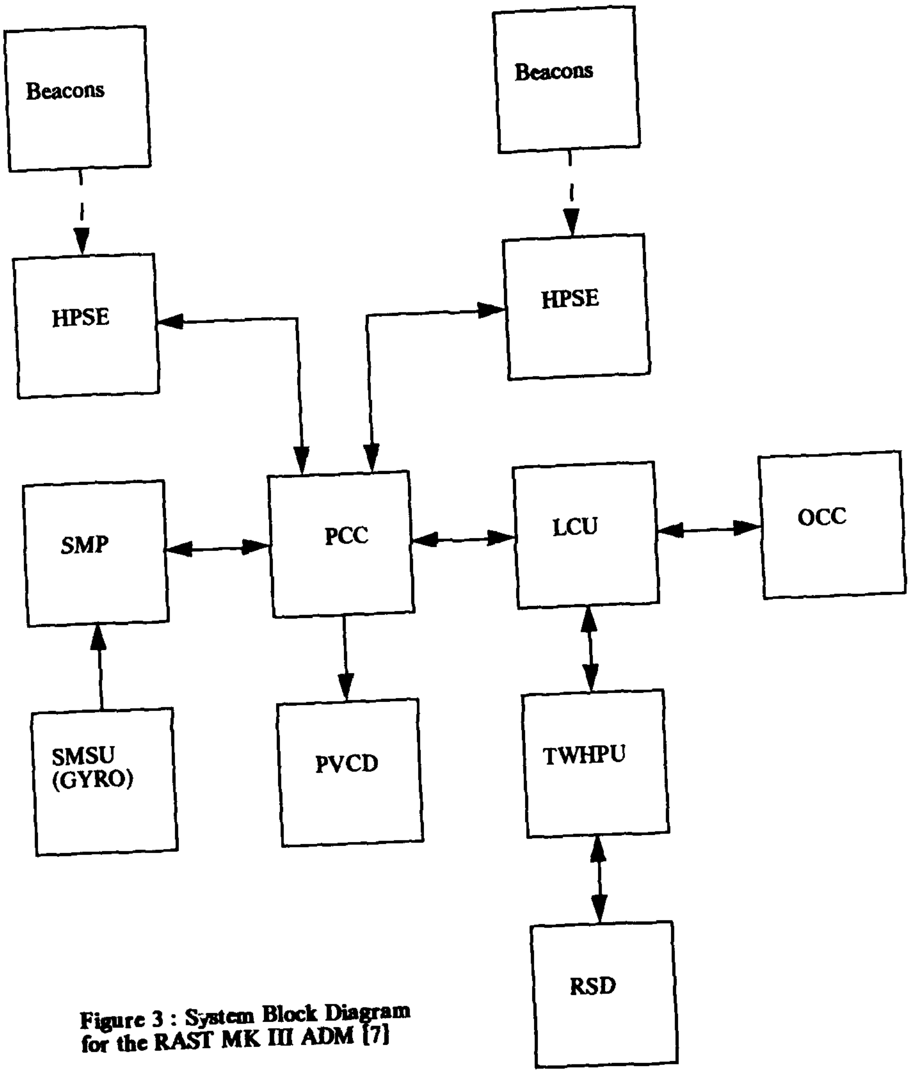


Figure 3 : System Block Diagram
for the RAST MK III ADM [7]

10.1 Rapid Securing Device / Traverse Winch & Hydraulic Power Unit

The Rapid Securing Device (RSD) provides the means of securing the helicopter to the deck. It is constrained to travel fore/aft in a deck track under the control of the traverse winch, and can also exert lateral maneuvering forces to straighten the aircraft. The Traverse Winch & Hydraulic Power Unit (TWHPU) is used to traverse the RSD along the deck to and from the designated landing area [7].

10.2 Helicopter Position Sensing Equipment

The Helicopter Position Sensing Equipment (HPSE) is the component of the system that measures the position of the helicopter prior to landing relative to the ship's deck (Fig. 4) [7].

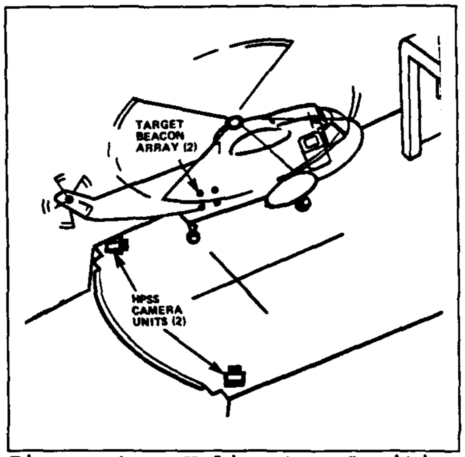


Figure 4 : Helicopter Position Sensing System [1]

10.3 Pilot Cues Controller

The Pilot Cues Controller (PCC) is the system controller. It utilizes the helicopter position data output by the HPSE to allow the traverse winch to move the RSD fore/aft to track the helicopter and also provides positional cues, and a landing prompt, to the pilot via the PVCD [7].

10.4 Pilot Visual Cues Display

The Pilot Visual Cues Display (PVCD) is a passive component that receives HPSE data and SMPE landing state prediction data as a basis for the generation of a landing prompt to the pilot via the PVCD. The PVCD is also used to display trafficator signals [7].

10.5 Ship Motion Prediction Equipment

The Ship Motion Prediction Equipment (SMPE) consists of the Ship Motion Sensor Unit (SMSU) hardware component and the Ship Motion Prediction (SMP) software component. The SMPE is used to monitor ship motion in six degrees of freedom, and from measured ship motion, predict ship motion for three or more seconds. Predicted ship motion is compared to anticipated helicopter motion to generate a landing prompt [7].

10.6 Local Control Unit and Operator Control Console

The Local Control Unit (LCU) serves as a central control unit and junction box which interfaces to the OCC, TWHPU, RSD, and PCC. It is designed to be bulkhead mounted inside the hanger. It has a limited number of duplicated manual controls to enable local control of the traverse winch for test and maintenance purposes [7].

11. SYSTEM OPERATIONAL OVERVIEW

The purpose of the RAST MK III ADM will be to assist helicopter pilots in landing on the flight deck of naval vessels. It will also perform the function of a conventional RAST system, that is securing, straightening, traversing the helicopter into the hanger and traversing the helicopter from the hanger to the launch position. The system will accomplish this in eight stages [8].

a. cameras mounted on the flight deck will acquire images of laser diode beacons located on the helicopter;

b. image processing of detected beacon co-ordinates will yield the position and orientation of the helicopter with respect to the cameras;

c. the helicopter position data will be transformed to the centre of the designated landing area, and the position of a probe mounted on the helicopter will then be calculated relative to the designated landing area co-ordinate frame;

d. the probe location will be compared to the position of the Rapid Securing Device (RSD), and this comparison will be used to generate a signal transmitted via the Local Control Unit (LCU) to maintain the RSD 500mm (+/- 200mm) aft of the projected probe touchdown point;

e. commands will be sent via a Pilot Visual Cues Display (PVCD) to indicate to the pilots whether the helicopter should move fore/aft, or port/starboard to bring the probe over the designated landing area;

f. ship motion in six degrees of freedom will be predicted several seconds into the future. Anticipated deck motion will be related to anticipated helicopter motion to provide data which will indicate safe or unsafe landing conditions;

g. the combination of helicopter probe location with respect to the designated landing area, along with Ship Motion Prediction safe-landing data will be used to generate a landing prompt signal to the helicopter pilot via the PVCD; and

h. after touchdown, the Landing Safety Officer (LSO) via the Operator Control Console (OCC), will move the RSD to secure the helicopter probe. The helicopter may then be straightened and traversed into the hanger.

12. SOFTWARE

The RAST MK III system software is being developed to a tailored version of DoD-STD-2167A (Defense System Software Development) and written in the Ada programming language (ANSI/MIL-STD-1815A). ITI is using a hybrid software development methodology on the RAST MK III ADM project. The structured requirements analysis technique as described by Page-Jones [9] has be used to graphically portray the data and control flows to and from each major component within each CSCI (Computer Software Configuration Item). These diagrams and techniques have become an integral part of each Software Requirements Specification (SRS).

Buhr diagrams will be used to further portray requirements, preliminary and detailed designs. Buhr refers to this design strategy as data-flow structured design [10]. This strategy identifies the key components of data flow in the system and then uses functional decomposition to identify transformation functions at nodal points in the data flow. The result is data flow graphs. The remaining step is to develop from this data a structure graph which describes the system control structures to implement the data flow. The strategy is applied consistently at configuration item, component and unit levels.

13. CONCLUSION

The Canadian Navy has realized that there was a requirement to investigate possible solution to the critically decreasing weight margins being imposed on ships. This has resulted in the requirement for lighter and more compact ship systems. The HHRSD system was one of the systems that was examined in an attempt to solve this problem. If the current HHRSD system was to be replaced, then the replacement system would be required to at least maintain the current operational envelope.

A concept feasibility study was commissioned that determined it was possible to at least maintain the current flying envelope while reducing weight, complexity and space of the system. Because of the minimum amount of associated equipment, and for example, the elimination of the current requirement for a separate helicopter hauldown compartment, RAST MK III will offer an overall reduction in volume and weight high up in the ship of up to five tonnes over the HHRSD system. Other significant advantages of the RAST MK III are:

a. RAST MK III offers an estimated savings of up to 60 percent when compared to the initial acquisition and overall life-cycle costs of the current in-service Canadian system. From a software perspective, this is one of the primary aims for the Defence System Software Development Standard (DoD-STD-2167A) and Ada. Other projects (i.e. the Advanced Field Artillery Tactical Data System (AFATDS)) have demonstrated that the heavy emphasis on requirements specification and design reduces the number of errors and, hence testing and debugging time [11];

b. the reliability and maintainability of the system will be improved, again due to the requirements of good software development practices and the reduction in associated equipment;

c. with the aid of the Position Sensing System and the Pilot Visual Cues system the helicopter can be quickly and accurately positioned, thereby reducing the time required in both the high and low hover positions. The elimination of the requirement for a hook-up procedure also reduces the recovery time; and

d. no personnel other than the LSO (Landing Safety Officer) are required on the flightdeck while the helicopter is hovering, landing, being aligned or traversed.

RAST MK III is a mature Research and Development project which will carry shipborne helicopter operations well into the next century. It is seen as the ideal candidate for retrofit on in-service ships (during refits or mid-life updates), and new ship programs.

14. ACKNOWLEDGMENT

The authors gratefully acknowledge the support and cooperation of Indal Technologies Incorporated.

15. REFERENCES

1. Cdr R. Johnson. "RAST MK III - A New Generation Helicopter Handling System". Maritime Engineering Journal. September 1989. pp.6-10.

2. Indal Technologies Incorporated. "Recovery Assist, Secure and Traverse System (RAST) Mark III - Concept Feasibility Study - Task 1 Report". March 1986.

3. Indal Technologies Incorporated. "Recovery Assist, Secure and Traverse System (RAST) Mark III - Concept Feasibility Study - Task 2 Report". May 1986.

4. Indal Technologies Incorporated. "Recovery Assist, Secure and Traverse System (RAST) Mark III - Helicopter Position Sensing Options Rev. 2". April 1988.

5. Indal Technologies Incorporated. "Recovery Assist, Secure and Traverse System (RAST) Mark III - Helicopter Position Sensing Equipment - Task 7". August 1988.

6. Minutes of Progress Review Meeting Held at Indal Technologies Incorporated in Mississauga, Ontario, 28 March, 1990.

7. Indal Technologies Incorporated. " System Design Document (SSDD) For The Advanced Development Model Of The Recovery Assist, Securing and Traversing System." February 1990.

8. Indal Technologies Incorporated. " System Specification (SSS) For The Advanced Development Model Of The Recovery Assist, Securing and Traversing System." February 1990.

9. M. Page-Jones. The Practical Guide To Structured System Design. Yourdon Press. Second Edition. 1988.

10. R.J.A. Buhr. System Design With Ada. Prentice-Hall Inc. Englewood Cliffs, New Jersey. 1984.

11. Donald Firesmith. "Ada Project Management Course". March 1990.

16. DISCLAIMER

The opinions expressed in this paper are those of the authors and, as such, are not necessarily endorsed by the Department of National Defence (Canada) and/or Indal Technologies Incorporated.

REUSABLE SOFTWARE WITH ADA:
A COMPETITIVE EDGE FOR THE '90s

by
James R. Jefferson and Roger S. Hebden
PDI CORP.

1. ABSTRACT

The defense software industry is faced with the difficult challenge of keeping pace with rapid advances in evolving hardware technology. Effective implementation of the functional performance requirements associated with modern defense systems must utilize innovative software development approaches aimed at improving productivity and quality, especially when facing the reality of anticipated defense budget cuts. A structured approach to software reuse with Ada is presented as a possible element to a solution in meeting this challenge. A software reuse strategy utilizing Ada offers an opportunity to alter the economic trend of ever increasing software life-cycle costs.

2. INTRODUCTION

The purpose of this paper is to describe top level considerations for establishing a software reuse strategy at the company level. The concepts to be presented here have evolved as a result of lessons learned related to software reuse on prior DOD projects, and represent our current approach to software reuse. Other software development organizations may wish to consider these concepts in establishing an internal plan to improve software productivity.

Truly remarkable advances have been achieved in the technology and manufacturing of computer hardware over the past decade. As a familiar example, the personal computer as we define it today was non-existent as recently as nine years ago. In the realm of embedded microprocessor-based controls, eight-bit processor based systems executing several hundred-thousand instructions per second were providing exciting new platforms for machinery control and data acquisition applications. The term

"MIP" (million instructions per second) was reserved for those huge and costly mainframe computers. Today, microprocessors are capable of executing in excess of 25 million instructions per second. Within two years, chip vendors expect that throughput to approximately double. The clear bottleneck in the productive utilization of this ballooning computational power lies with the inability of the software development process to keep pace. As the hardware cost/performance ratio continues to decrease, the software development process consumes a growing percentage of system development costs.

3. THE PROBLEM

The fundamental challenge facing the software engineering industry today is one of economics. The development of large scale software systems has become expensive. As the current decade opens, limited supplies of engineering and fiscal resources present a significant hurdle in delivering successful software systems. Available resources must thus be leveraged to support the increasing demands associated with the development of complex software applications.

This challenge, often referred to as the software engineering crisis, is not new to the industry. Attempts to improve software productivity have been made in the areas of high order languages, Computer Aided Software Engineering Tools (CASE) and software analysis and design methodologies. Although each of these elements has contributed significantly to advancing the state-of-the-art in software development, the problem is more deeply seated than one which may be solved by the development of a new language, tool or set of design procedures.

The essence of the problem is centered in the manner in which software systems are built. That is, software developers have generally taken the short-term perspective resulting in brittle designs which meet only specific project requirements rather than flexible designs intended to be reused in future applications. With each new software system, developers often redesign and re-implement every component within the system, frequently without knowledge of any overlap with previously designed systems. This often results from the independence of multiple design teams through the compartmenting of the software organization along program specific boundaries. In contrast, such an approach is unheard of in the electronics manufacturing industry. The essential premise of manufacturing capitalizes

upon the idea of utilizing off-the-shelf parts to meet changing requirements rather than redesigning and building existing parts. The success in the development of computer hardware can, in part, be attributed to the ability of the industry to build upon itself in the manufacturing of new and more capable products. The computer hardware industry has enormous flexibility with respect to the sheer number of components available and the ease with which they may be purchased. These components range in complexity from primitive level chip sets to complex computer systems. Each component, in turn, may be used or thought of as a part of a new and larger system.

In the creation of software systems, developers have failed to realize such levels of efficiency. Unfortunately, the software development process is not characterized by the efficient utilization of parts catalogs, but rather by the re-creation of parts already working in other applications. The challenge is to change the way software development is viewed by drawing upon the premise of off-the-shelf parts inherent in more mature disciplines such as electronics engineering. In short, the software industry must learn software manufacturing.

4. REUSABLE SOFTWARE - A PROMISING SOLUTION

It is our belief that the requisite tools and software engineering principles are currently in place to enable significant improvements in software development productivity. Software reusability, which has already seen large scale returns in Japanese software factories (1) represents a viable course of action that can be undertaken individually by software development organizations today. From a national perspective, software reuse represents an opportunity for the software industry to adequately support the development of new sophisticated defense systems given the limited resources available. From a corporate perspective, the successful implementation of a software reuse strategy will be an essential ingredient for remaining competitive in the 1990's.

4.1 Why software reuse?

Software reuse should be pursued for its intrinsic long-term productivity and economic merits. Results of initial efforts in software reuse at PDI in the shipboard propulsion plant training simulator domain are summarized below. These efforts involved the development of a series of propulsion plant

operator trainer systems beginning in early 1987 with the first trainer completing factory acceptance testing in April of 1989. Development of a second functionally enhanced trainer system, which utilizes a design derived from the first, was begun in mid-1989, with the software anticipated to be completed in later this year. Both systems are implemented entirely in Ada using an object-oriented design methodology.

- o Costs can be reduced and productivity increased when developing systems derived from successful previous designs. Relative costs associated with these systems are as follows:

	Base System	Derived System
Source lines of code	50,000	65,000
Avg. cost per tested source line of code (in man-hours)	0.86	0.31
Productivity (avg. number of tested source lines per man-day)	9.3	25.4

  As reflected in the numbers above, the development of a new system based on a previous design resulted in a software productivity increase of over 273% as measured in terms of source lines of code per man-day. Stated in other terms, this translates into a 63% savings in software development costs.

- o Commonality is the key to reuse. The highest probability for a good return on an investment in software reuse occurs when well defined and focused application areas are targeted for development of reusable components.

- o Reuse of applicable previous requirements promotes reuse of previous designs and code.

- o Designs and specifications as represented in a program design language or in an object-oriented graphic format provide an important class of reusable component. The reusable design or design template is an especially

potent reuse form because it allows flexibility in the final implementation of the resulting code. In contrast, highly integrated reusable code components are often too specialized for verbatim incorporation.

- o Use of the Ada programming language and an object-oriented design approach provide a method in establishing natural encapsulating boundaries defining the scope and interface to the reusable component.

- o By identifying candidate reusable software components before they are actually designed and implemented, the probability that the component will actually be reused in the future is greatly enhanced since adequate design attention can be applied in ensuring component interfaces are of more generic nature.

The method of reuse applied in the design of the propulsion plant trainer systems described above consisted of a simple process of "harvesting" completed software components resident within the original design. Since the application mission of the trainer systems is very similar, many objects defined in the first design were equally applicable to the second. It was discovered however, that the method utilized to define reuse components did not lend itself well to the creation of an organized company reuse repository. This "harvesting" method of defining the contents of a reusable software library is analogous to a bottom-up reuse design strategy in that the scope, architecture and application coverage of the contents within the reuse repository tend to haphazardly wander without direction as completed projects coincidentally provide component contributions to the repository.

Based on the lessons learned, our thinking and approach to software reuse has evolved into a top-down method. In using this approach, the reuse repository architecture and the anticipated or desired components required to fully support the defined application area are defined well in advance of the actual component development. Since commonality in the application context is the key to productive reuse, the very first step in designing this reuse repository is to define the specific application domain being addressed. A domain is defined as any recurring technology based application area in which software systems are utilized as part of the implementation. As described above, one such application domain has been defined as the shipboard propulsion plant training simulator domain.

4.2 Domain reuse, technical issues

The creation of a domain-oriented reuse repository requires three basic phases of activity: domain analysis, domain development, and domain control. The purpose and activities associated with each of these phase is described in the sections that follow.

a. Domain analysis Domain analysis is the activity during which the scope and focus of a given application domain is defined. The product of this analysis is a domain specific taxonomy or classification system depicting the elements and associations which make up the domain. This taxonomy, which effectively represents the top level design in the software reuse strategy, can be easily visualized as a family-tree fashioned breakdown and ordering of components anticipated to have reuse value and generic application to future system development. "Domain analysis is important to reusability [in] that it forms the basis for creating reusable components. Instead of building an ad-hoc collection of software components, components should be built that encapsulate common objects and operations identified by domain analysis. Such an approach substantially increases the reusability potential of a software collection." (2)

In the development of a domain specific taxonomy, an analysis methodology must be rigorously followed, thereby establishing a solid foundation upon which the domain analysis effort can be performed. In recent years, object-oriented development (3) has received substantial attention and appears at present to provide the most promising analysis perspective in establishing such a foundation. Object-oriented development is a software analysis and design methodology based upon the concept of partitioning a problem solution into objects and the associated actions performed on or by these objects. In using this perspective, object-oriented development encourages a simplified approach to software development by producing solutions which resemble the "real world". The object-oriented viewpoint reflects the natural structure of the problem domain rather than the implicit structure of the process or data underlying the problem; thus it provides a more understandable approach to software development.

Object-oriented development supports the goal of domain analysis by allowing the analysis to utilize a well understood and industry accepted methodology. The principles associated with this method support the decomposition of an application domain through the identification of objects within the domain and the subsequent classification of those objects according to common characteristics. This process leads to the definition of the domain specific taxonomy. Within the propulsion plant trainer domain, the LM2500 gas turbine engine simulation as utilized on the DDG 51 and previous ship classes is clearly identifiable as an object. Furthermore, the LM2500 object may be thought of as being composed of many smaller objects or subsystems. Items such as the ignition subsystem and fuel control subsystem are all components of the LM2500 and can also be identified as objects in the decomposition and analysis of the engine simulation. In each of these cases, the objects are identified by a certain set of characteristics and relationships with other objects. Yet, in the complete analysis of the domain, other non-physical objects must also be addressed. For example, abstract entities such as terminal/screen handlers, simulation casualty control, communication controllers, peripheral device handlers, and hardware specific interface device drivers are all essential objects within a real-time trainer.

Classification is the activity of grouping similar objects based upon common characteristics, operations, attributes and relationships. All members of a group or class share one or more characteristics that members of other classes do not possess. For example, a gas turbine engine shares certain characteristics with a that of a diesel engine. Both include characteristics such as fuel intake, combustion and torque and could be classified accordingly. By factoring out the unique characteristics, the common characteristics of a generic engine object are discernable.

Once the initial taxonomy has been defined, it must be incrementally updated as the domain scope or technology changes. In addition, the complexity of the taxonomy may evolve as knowledge of the domain increases.

b. Domain development Domain development is the activity associated with the actual design and development of the reusable software components identified through the previous analysis process. These components are created through development efforts in which the components are designed and coded as either

a by-product of on-going development programs or as selected research and development tasks. The domain taxonomy provides a means by which the development process is directed. This critical guidance ensures that the reusable software components are well focused and generically applicable.

The actual development effort must be based upon established common directives. Standard practices such as utilization of a common language which supports reusability, and adherence to a set of development guidelines must be established in advance. Ada, as both a program design and software source language, provides the necessary resources upon which a working set of reusable software components can be developed. These resources are in the form of syntactic and semantic constructs that help standardize the creation of reusable software. The language features which promote reusability include the Ada package, generics, strong typing and overload resolution. In addition, the endorsement of Ada as an ANSI standard (4) along with its mandated use by the U.S. Department of Defense increase Ada's utility for reuse and further enhance the probability that Ada will provide a significant impact on software productivity.

Reusable software components may assume one of several forms. Reusable code is the most familiar and accepted form of reusable software and consists of complete source or object modules for general purpose usage across various applications. Math functions, data structures and sort routines are frequently encountered examples. These components support the development of software systems by providing the low level building blocks through which new systems may be developed. From this perspective, reusable code is a critical element in formulating any reuse strategy.

However, reusable code in itself is limited in its ability to significantly impact the software development life-cycle outside of the detailed design and coding phases of software development. Reusable design, on the other hand, provides a more powerful form of reuse through the encapsulation of broad design decisions and technical approaches. For example, within the propulsion plant operator trainer domain, the architecture of the generic engine model is a prime candidate for capturing as reusable design. A design such as this could be selected for reuse early in the development of a software system thereby significantly impacting the direction and momentum of the project. In applying reusable design, the goal is to improve

developmental efficiency by eliminating redundant design effort associated with the building of similar systems. These reusable designs typically exist in the form of program design language or code templates representing a specific design of an object or class of objects. The reusable design may be thought of as the blueprint which maps architectures to recurring problems.

The actual process of domain development consists of populating the previously established domain taxonomy with completed reusable software components. The priority and order of development should target those objects that offer the most potential for reuse. Those objects are implemented first since they offer the quickest anticipated return on the reuse investment.

c. Domain control Domain control refers to the activities associated with the storage and retrieval of reusable software components. These activities are centered around a reusable software library. A reusable software library is a repository, usually in the form of a database or information retrieval system, which contains the reusable software components developed as a result of the domain development effort. These components are represented by reusable requirements, design templates and source code.

The purpose in establishing a reusable software repository is to provide a means to access and evaluate reusable software components for possible applicability to new product development. Design team members must be aware of a component and its capabilities before they can consider it for reuse. All libraries are fundamentally based upon an underlying classification system. In the development of a reusable software library, this classification system draws upon the taxonomy generated during the domain analysis effort. This taxonomy provides the backbone of the reusable software library by supplying the subject categories, topics, keywords and overall structure through which components may be referenced.

Configuration control of the reusable software components library is probably the most critical factor in ensuring the long term success of a reusable software strategy. It is through this control that the integrity and overall quality of the reusable software components is maintained. These controls should at a

minimum incorporate a reuse configuration review board responsible for approving candidate objects to be subjected to a development effort, and for final design and code approval of the finished component prior to formal entry into the reuse library. Additionally, a formally designated reuse configuration manager must be in place to administer and maintain integrity of the library. Libraries littered with non-generic, brittle one-time use components will lose the endorsement and support of the engineering design groups, frustrated in the search for applicable generic components. The quality of the reuse program can be no greater that the quality of the components within the reuse library. These formal procedures are thus established to ensure that elements entering the library are of the highest possible quality since the goal is to utilize these components as the foundation for all future work within this application domain.

d. Putting it all together Software reuse achieves the greatest economic impact when opportunities for reuse are factored into the early phases of system level definition. During this early stage, system requirements are allocated to hardware and software configuration items. With full knowledge of the pre-existing software architectures and design templates available from the reuse repository, system engineers with software engineering support may intelligently steer this requirements allocation process in a manner that:

- o most fully utilizes pre-existing and tested architectures, design templates, and source code components, and
- o encapsulates requirements for candidates of new reusable components to be developed as a part of the current design process in order to further expand the repository.

As indicated in Figure 1, which illustrates a summary of the domain reuse process, the next opportunity to exercise reuse occurs during the software requirements analysis and top level design phases where software architectures and subsystem designs are examined. In a similar manner, but at a lower level, software designers of a given subsystem again look to the repository for design templates and code components when performing the detailed design activity. When the coding phase is finally reached, software implementers look to the repository for reusable device drivers, low level utilities, data structures, and math library routines.

4.3 Domain reuse, non-technical issues

The software reuse process must be viewed as a long term investment since little payback occurs during the early evolution of the software repository. In fact, this lack of immediate payback is the source of a frequently utilized argument against reuse since program managers of individual programs have little incentive to invest in the future of other projects at their expense. After all, their individual performance evaluation is based on meeting costs and schedules associated with their

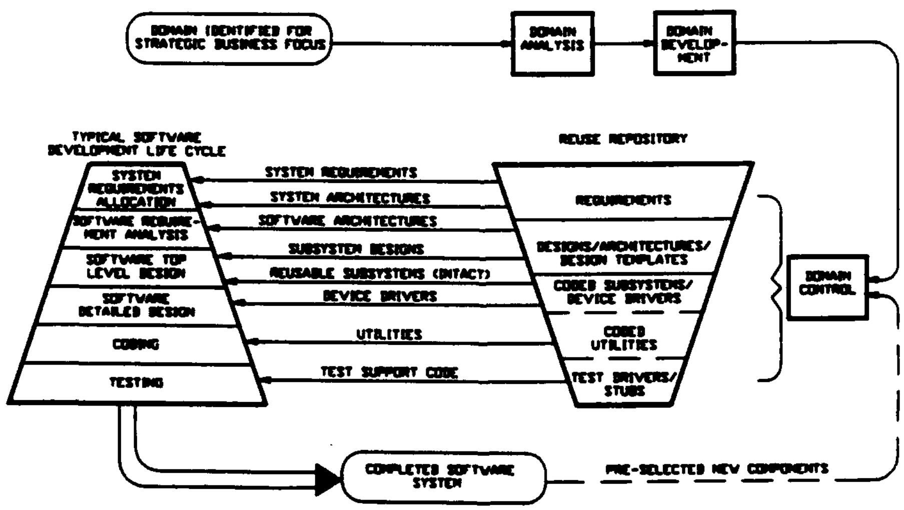


FIGURE 1
DOMAIN SOFTWARE REUSE

assigned program. Both corporate management and DOD contracting agencies need to understand the strategic role of a company software reuse plan and provide the necessary motivation and incentives rewarding its application.

Since effective reuse requires consideration at all phases of the design including front-end system definition, appreciation of the reuse process must be instilled as a corporate culture across all engineering disciplines. Having defined a workable method for integrating reuse engineering into all projects is not sufficient to ensure the reuse process will ultimately succeed. A corporate-wide understanding and commitment to the goal is essential. This in turn requires an ongoing cross-discipline familiarization program in available reuse technology, and a project organization that clearly assigns and defines reuse engineering responsibilities at all levels. Reuse engineering activities at each phase of the design are thus assured of gaining the requisite visibility and priority in balance with the traditional design activities.

5. SUMMARY AND CONCLUSION

Substantial work is currently evolving in the realm of software reuse as its potential for economic benefit is gaining recognition. STARS (5), one of the most noted government sponsored reuse programs, has been involved with the development of a national network of reusable software repositories. However, general availability of these repositories is years away, with many complex legal and social issues still pending resolution. Concerns over product liability, client organization endorsement, logistics problems associated with the dissemination of repository contents, and a reluctance of developers to place company proprietary technology into the public domain must all be overcome. From the STARS repository user's perspective, problems such as the "not invented here" mentality, concerns over software quality and adequacy of testing, difficulties in keeping current with the global repository contents, and the problem of simply finding those few components that have relevance to the specific domain of interest all present a challenge against which the benefits of such reuse must be weighed. However, with an organized reuse program maintained and managed at the company level, each of these problems can be solved today. The Government sponsored STARS program may someday significantly impact the entire process of Government software procurement through mandatory utilization of pre-specified software

components invoked at the contract level. In the meantime, organization internal reuse focused within traditional market niches offers a more immediate solution in maintaining a competitive edge.

Significant leverage in software development resources can be achieved today through the utilization of software reuse methods managed at the company level. The reuse strategy that is currently evolving at PDI CORP. consists of the following key elements.

- Software reuse efforts must be organized and bounded into highly specific and cohesive application domains. Commonality is the key to obtaining economic returns from reuse.

- Each application domain must undergo a development cycle that parallels the modern software development cycle. Domain analysis is analogous to the top-level architecture design phase. Domain development is analogous to the design/coding/test phase. Domain control provides the essential configuration management function.

- Ada is being used as the basis language. This language's formal specification, Government endorsement and natural support of the object oriented development approach make it highly suitable.

- An object oriented development method is utilized. Just as in the physical world, where real objects are reused in different combinations with other objects, the same can be true of software if designed from the object perspective.

- The reuse strategy must be understood and supported by both the engineering organization and corporate management.

The definition of software reuse is much broader that simply the process of looking for opportunities to incorporate previously coded subroutines into a current development project. Those organizations who view software reuse as the inclusion of a few utilities or simple re-application of pre-established data structures are missing a much more rewarding opportunity.

Narrowly defined views of reuse have produced only limited returns. The key to the realization of the true economies offered through software reuse is through large scale reuse spanning all phases of the software development cycle.

7. REFERENCES

1 Tracz, W., "Software Reuse: Motivators and Inhibitors", Proceedings of the COMPCON '87. IEEE Computer Society Press, Washington D.C., February, 1987 reprinted in Tracz, W., "Software Reuse: Emerging Technology", Computer Society Press, Washington, D.C., 1988.

2 Prieto-Diaz, R., Jones, G.A., "Breathing New Life into Old Software", GTE Journal of Sciences and Technology, Vol 1, Spring 1987 reprinted in Tracz, W., "Software Reuse: Emerging Technology", Computer Society Press, Washington, D.C., 1988.

3 EVB Software Engineering, Inc., "Object Oriented Development for Ada Software", The Benjamin/Cummings Publishing Company, Inc. 1987.

4 United States Department of Defense, "Reference Manual for the Ada Language: ANSI/MIL-STD-1815A-1983", United States Department of Defense, February, 1983.

5 Office of Secretary of Defense, "Software Technology for Adaptable Reliable Systems - STARS: Technical Program Plan", Office of Secretary of Defense, Washington, D.C., August 1986.

SURVIVABILITY OF THE PLATFORM UNDER COMBAT DAMAGE

by

LASZLO B. MAYER

SECURIPLEX, POINTE-CLAIRE, CANADA

1.0 ABSTRACT

Modern warships rely primarily on active electronic counter-measures to eliminate inbound threats before they can strike. Having chosen firepower over defensive power, these ships are lighter than their heavily armored World War II predecessors. As a result, these ships are also more vulnerable to catastrophic damage (1). Experiences with personnel casualties and extensive shipboard damage caused by hostile actions have demonstrated the need to improve the present approach to the design and implementation of naval damage surveillance and control systems.

Consequently, there is an urgent requirement for the development of a survivable, reliable and maintainable damage control system capable of supporting real-time, informed decision making.

During this development, the concept of damage control must focus on situations realistic during war-time action. The damage control system must maintain operation after the platform has sustained damage effected by mines or missile explosions, and resulting in fire, flood and structural damage. Attention must be given to more than the failure of sensors or wiring loops. Emphasis must be placed on recovering from damage to any system component and the implementation of system reconfiguration.

Successful performance of real-time damage control in large and complex surveillance and control systems is not achievable by operators alone. As these systems become more and more complex, there is a need for knowledge-based expert systems to assist the operator in the form of automated decision aids and/or corrective actions.

This paper presents a critical assessment of present-day damage control technology, and looks forward to future systems, outlining approaches which offer the greatest potential for ensuring the continued availability, mobility and survivability of the fighting platform in a combat environment.

2.0 INTRODUCTION

The fundamental operational objective of a damage control system is to ensure the timely and informed application of the ship's resources to the containment and control of damage. The achievement of this objective requires the monitoring and surveillance of the ship's integrity prior to sustaining damage; the reporting of damage to the damage control organization and the co-ordinated, directed response by the damage control organization in a timely and efficient manner.

In most in-service warships, surveillance depends on a number of independent monitoring systems for fire, flood, etc., and the ability of damage control roundsmen to detect and report abnormal conditions. The information is manually collated and plotted on state boards. This methodology for assimilating critical data for the containment and control of damage is primarily a manpower intensive operation originally devised during the last two World Wars.

The decision makers are left without an accurate description of the damage, impeding the prioritizing and coordinating of control efforts at the scene of damage. During recent naval conflicts (2), (3), fires have spread out of control while damage investigation and control was attempted under conditions of toxic smoke, extreme heat, flood, jammed access-ways, etc.

The new Canadian Patrol Frigate (CPF) currently being built incorporates an extensive damage surveillance and control system. It consists of the following four subsystems:

a.) **Fire Detection and Suppression Control**, dividing the ship into 75 damage control zones, extensively monitored by smoke, heat and explosive gas detectors. The suppression capability relies on the combination of a remotely actuated AFFF system and manually/automatically actuated Halon-1301 systems for 36 electronic and high-value spaces.

b.) **Firemain Status and Control System**, providing an overview of the ship's firemain along with duplicated centralized control capability over the isolation valves, fire pumps, prewet, sprinkler and magazine deluge systems. Bilge flooding alarms are also displayed for the operator.

c.) **Ventilation Status and Control System**, allowing the automated sectioning off of the ship's ventilation system prior to extinguishant release, along with the selection of smoke eduction configurations. Additionally, the computerized configuration of the ship's ventilation system for the establishment of gas-tight citadels for NBC protection is provided for.

d.) **Liquid Level Management System**, facilitating the centralized control and *monitoring of the ship's* diesel and helo fuel, fresh and ballast water tanks. Remote manual control is provided for tank valves and transfer pumps to permit counter flooding and weight re-distribution.

The overall CPF damage control system monitors and/or controls approximately 1000 points distributed homogeneously throughout the vessel. The damage control team is provided with an overall global summary of the ship's integrity, with an indication of any off-normal condition within one-second of occurence, via hard mimic isometric panels. Control is exercised by the coincidental activation of a mimic illuminated pushbutton (to select location) and a physically separated executive pushbutton (to select desired function). The light within the mimic pushbutton provides monitored status information.

This system is a significant improvement over those previously fitted systems in that it provides the decision maker with an increased capability to monitor real-time sensor status and remotely control essential equipment. The control system, due to its interactive configuration, will ideally support the assessment and control of localized minor damage when alarms are activated.

However, the development and implementation of a real-time, informed strategy for the containment and control of major battle damage, caused by torpedo or missile impact resulting in a multitude of fire alarms, flooding, and extensive structural damage will become unmanageable. The decision maker will be saturated with the volume of data. He will be expected to observe and correlate several alarm panels, and to understand numerous voice reports encompassing dozens of spaces. Without knowing which reports are current or reliable, he will be expected to assimilate the incoming damage reports, determine the optimal overall strategy, and to make immediate decisions and give orders to set damage boundaries, section off ventilation systems and isolate firemain sections.

In summary, even though the CPF ship design incorporates the most up-to-date resources available to detect and engage damage, it is difficult and takes too long for the damage control decision makers to interpret reports and assess the extent of damage, especially under conditions of exceptional stress.

Clearly, further development is required to improve the performance of the damage control system that will enable it to cope with realistic war-time damage. The development work must specifically address the issue of rapid decision making, thereby providing the ability to exercise coordinated, real-time control of men and equipment at the scene of damage.

3.0 DEVELOPMENT OBJECTIVES

Given that present-day ships are built with excellent damage engagement capabilities, the specifics of the mechanical systems will not be addressed as part of this paper.

The objective is to concentrate on damage control in terms of evaluation and decision making - that is on the management of damage - and focus on providing the capability for the decision maker to develop and implement a rapid, informed strategy for the assessment and subsequent containment and control of damage.

More specifically, this paper will outline the key steps involved in the process of developing a knowledge-based expert advisory system, and defines the interfaces to be used to communicate the information to the decision maker.

The desired operational capabilities of the required system are the following:

a.) Continuously monitor sensors for fire, flood, citadel breach, etc., and update critical system status as they affect damage control operations.

b.) Detect and diagnose problems.

c.) Postulate real-time recommended damage control strategies based on proven naval doctrines, to minimize damage.

d.) Provide structured, integrated information to maximize the decision maker's effectiveness.

e.) Provide embedded damage control training.

The subsequent sections of this paper provide an overview of the organization and development process of a knowledge-based system, along with the constraints and requirements imposed on the architecture and man-machine interface for an expert system assisted damage control system.

4.0 KNOWLEDGE-BASED EXPERT SYSTEM

4.1 Introduction

In simple terms, an expert system is a computer-based system that uses knowledge, facts, and reasoning techniques to solve problems that would normally require the expertise and ability of a human expert.

The notion behind exploring the implementation of an expert system for damage control is based on the following conditions faced by the damage control decision maker in a realistic battle situation:

a.) Decision making involves assimilating and prioritizing a large volume of unstructured data.

b.) It is necessary to reason with uncertain and possibly erroneous data and make a large number of judgement-based decisions.

c.) Experienced, rapid decision making is required despite lack of sufficient training under stress conditions.

Expert systems can monitor, interpret, diagnose, plan, schedule, control and train in order to provide more effective problem solving and decision making.

The expert system concept has been specifically developed to capture the problem solving expertise of a human being, and represent this person's knowledge in a data base in such a way that the computer can approximate the expert's ability to solve a problem.

The expert is someone who has developed more knowledge in a particular subject than most people in the same field, and who can use that knowledge to work with superior efficiency and effectiveness. Experts get to be experts through a combination of training and experience. Training gives them facts; experience gives them principles and hunches to use in applying those facts. It is this combination of facts, principles and hunches that is identified as expert knowledge. An expert can solve problems based on incomplete information using heuristics (informed hunches, educated guesses and rules of thumb) to fill in the gaps. Developing a good set of heuristics is a necessity for building an expert system (4).

4.2 Developing a Knowledge-based Damage Control System

Despite all the excitement surrounding artificial intelligence and expert systems, these systems are nothing more than computer programs operating on a set of data bases. Consequently, some standard software engineering methods are equally applicable to expert systems. Software lifecycle processes are also present in expert system development. The development of the expert system consists of the following phases:

4.3 Feasibility Study

There is not a single way of determining the feasibility of an expert system application, but some steps to be followed in the evaluation are crucial. One of these is gaining familiarity with the capabilities of current tools and the type of reasoning that are possible. Another is a detailed analysis of the application, and the third is a careful analysis of exactly how the final product is to be used and precisely what it will be expected to provide for the user.

The purpose of the feasibility study is to determine whether the implementation of the expert system based damage control system is technically and economically feasible.

Such technical issues as the appropriateness of the expert system versus conventional software technology; the availability and interest of suitable expertise; the form, size and complexity of the knowledge, and the methodology for the eventual system validation and testing must be examined in detail (5).

The economic analysis must estimate development and production costs, user benefits derived from the product, and market potential.

In order to permit an accurate assessment of the above technical and economic issues, it is essential to develop a preliminary "conceptual" design of the system. This design can be restricted to a scaled down, thus manageable portion of the application that is representative of the concepts and structure of the eventual expert system.

During the feasibility study phase, a review of available expert system shells must be made. These software development packages have specifically been designed to enable the rapid development of expert systems by focusing on the application rather than on the programming.

4.4 Designing of the Expert System

The actual design process consists of two distinct but interrelated activities, leading to the specification of the format of the knowledge representation and the development tools.These activities must be conducted in parallel, but not independently, as decisions about one will influence the other.

4.4.1 Knowledge Representation

The knowledge base is the key component of an expert system since it contains the expert's factual and relationship knowledge about the application. There are two broad categories of knowledge representation: rules and frames. Rules establish a relationship between facts in the knowledge base, conceptually represented as IF/THEN statements.

Frame-based representation provides a mechanism for structuring related facts in a data base.

Knowledge about properties, characteristics or behavior of objects, including permanent relationships (which are essentially properties), are best suited for frame-based representation. Knowledge about dynamic relationships between facts are best expressed in rules.

4.4.2 Development Tools

In principle, any computer programming language, even assembly language, could be used for the development. However, practicality dictates the use of some high-level language. LISP and PROLOG are the programming languages most often associated with artificial intelligence applications, but systems have been written in Pascal, Fortran and C.

In addition to the general purpose programming language systems, there are a wide variety of expert system shells. These are comparable to the multitude of word processor and data-base shells that aid in the creation of text and data files. The expert system shell is a reasoning system into which the application data is entered to create the expert system.

4.4.3 Selection of the Representation and the Development Tools

Damage surveillance and control requires the continuous monitoring and checking of various sensors, actuators and equipment configuration. The incoming data is interpreted, prioritized, and corrective action is recommended (or taken) by the expert system. This relationship knowledge is best represented and reasoned with in a rule-based structure.

Many different rule-based expert system shells are available, set apart essentially by the direction of their reasoning process. Some reason forward from the available facts, and by deducing new facts, arrive at a consequence. Some reason backward from a hypothesized consequence and try to locate known causes that will provide support. In summary, forward reasoning predicts; backward reasoning discovers. Some shells are able to reason both forward and backward depending on the nature of the problem and the information available.

The expert system in the damage control application will be required to reason forward from the monitored data to the optimal corrective action. If sufficient data is not available or trouble-shooting becomes necessary, the system will be required to reason backward from the known end-result and locate the cause. Consequently, an essential attribute of the expert system shell is the ability to reason in both forward and backward directions. If multiple solutions are identified, the expert system must be able to select the one with the highest

probability of success. Thus the ability to reason with uncertain data offered by some shells is also required.

A concern with the decision to go entirely rule-based is that it may involve an extensive set of rules, each relating to some sensor or equipment state. Should the run-time become too long, a shell, providing prioritization and multiple context reasoning, will be necessary.

4.5 Knowledge Acquisition

This phase deals with the task of obtaining knowledge from the experts, representing and structuring it for storage in the knowledge-base.

Since the expert system is only as good (or bad) as the expert who formulates the rules, the selection of the experts and the interviewing method are keys to the success of the product.

The means of obtaining knowledge from experts is through personal interviews starting by asking the expert for a superficial overview of assessing a damage scenario. The initial questioning should request the expert to describe the overall strategy, including the methods for categorizing the information used for reasoning. The knowledge obtained will not only identify the critical information required by the knowledge-base, but will also help in the design of the man-machine interfaces.

As the interviewing progresses from a superficial speculative level to specifics, the questions will also be specific, such as what does a piece of information mean, how does the expert use this information, and what he would do next and why?

Once the essential building blocks have been identified, they can be prioritized based on frequency of use and relative importance.

The solution methodology can now be evaluated, expressed in IF/THEN rules, and structured in the knowledge base.

5.0 SYSTEM DESIGN AND INTEGRATION

5.1 Introduction

The effective implementation of an expert system supported damage control technology into a warship environment must extend beyond the consideration of the mechanics of the actual interface compatibility. The architecture of the sensor/controller data networks, the data processing and user interface facilities, in addition to the actual integration and system validation must be examined.

5.2 System Architecture

The overall damage control system consists of the primary consoles containing the data processing and display units, secondary localized control and display units, data busses, sensors and controllers, in addition to the fitted fire fighting and damage control equipment.

The development of an optimal system architecture is driven by survivability objectives, and is constrained by the location and distribution of sensors, controllers and data processors.

The homogeneous distribution of sensors and controllers throughout the entire ship dictates that the type of data bus architecture used by a ship's machinery plant is inefficient and unsuitable for the damage control system. Furthermore, consideration of the requirement for the damage control system to be functional after major battle damage necessitates a significantly more survivable architecture for the damage control and communication system.

In order to maximize ship survivability and prevent system level unavailability caused either by damage or equipment failure, the optimal system architecture requires the sectioning of the ship into several zones for damage containment and control purposes. Although this architecture is independent of any specific ship class, for a frigate/destroyer class vessel this will typically require from 5 to 8 sections. Each section, as shown in figure 5.2, will require a pair of Local Control Units configured to operate in a dual redundant manner. Each of these units powers, monitors and controls the sensors and actuators in their respective section over a failure-proof data bus. Such a data bus is extensively used in a cost-effective manner by the CPF Damage Control System.

The individual Local Control Units are interconnected by a fiber-optic ring network, such as SAFENET (6), providing for the centralized monitoring, control and display of the entire ship's damage control status.

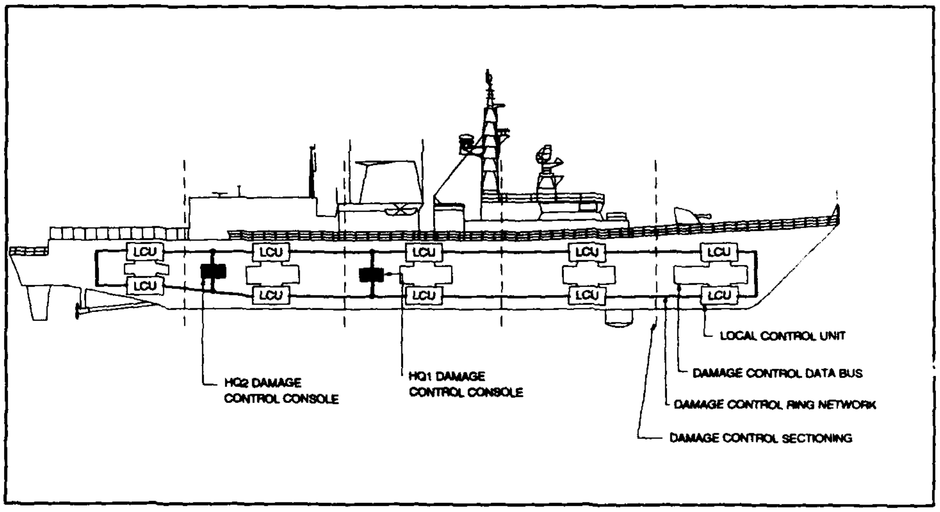


Figure 5.2 Representative Damage Control System Architecture

5.3 Operator Interfaces

The operator interface, or man-machine interface, is a critical factor in determining the underlying system's effectiveness. A knowledge-based system with little knowledge will not be viewed favorably, regardless of the interface design, but even a superbly "intelligent" system will be ineffective and unacceptable if the user interface is awkward or difficult to use. A project must devote approximately a third of its resources to the development of the man-machine interface (7).

During the design of the man-machine interface, the following questions must be kept in focus:

a.) Does the representation facilitate user understanding and reasoning?

b.) How efficiently (rapidly) can the user implement decisions?

c.) Is the information provided clear, concise and unambiguous?

d.) Can the user find and access needed information?

e.) Does the interface effectively provide system overview as well as specific priority details?

f.) How much training is necessary before the user can employ the system productively?

The graphic mimic panels such as those used by the CPF Damage Control System satisfy the above criteria, provided that low intensity data, characteristic of minor damage is required to be displayed. However, such an interface by itself becomes inadequate in the presence of any significant battle damage, and must be supplemented by visual display units capable of providing both text and image. To be effective, the design of the various displays must exceed the capabilities of the mimic panels in terms of information integration and clarity of representation. Accordingly, it would be a mistake to merely replicate the capabilities of the mimic panels in a different media.

The complexity of the information presented requires structuring, with the interrelationships between systems such as fire detection/ventilation status/fire suppression clearly defined. The nature of the damage control information presented is both textual and graphical. Thus windows are an essential feature of the display pages. Other essential features include text and image highlighting, and modification of the size, shape, color and texture of displays. Scrolling is additionally desired to transfer to adjacent CRT pages.

As the operator is not required to input a wide array of data, keyboard interface is not required. Instead, a menu-driven customized soft-key interface shall be used.

A conceptual damage control console for a frigate/destroyer class vessel is shown in figure 5.3, consisting of two CRTs along with a reduced ship isometric mimic panel. This arrangement retains the desired "overview at a glance" feature provided by the present CPF damage control mimic panels. The lighted portion of the mimic pushbuttons is intended to rapidly draw the operator's attention to any developing off-normal condition requiring investigation. The actuation of a mimic pushbutton, in coincidence with a CRT select pushbutton, will allow a rapid, structured and controlled access to positionally diverse CRT pages.

Finally the image and text presented to the operator, that is the delivery interface, is required to be "user proof". There must not be any traces of the underlying development interface during normal usage. An error must not place the user under system control or produce error messages from the underlying development system. The only interface the user should see, under any circumstances, is the specially constructed application interface.

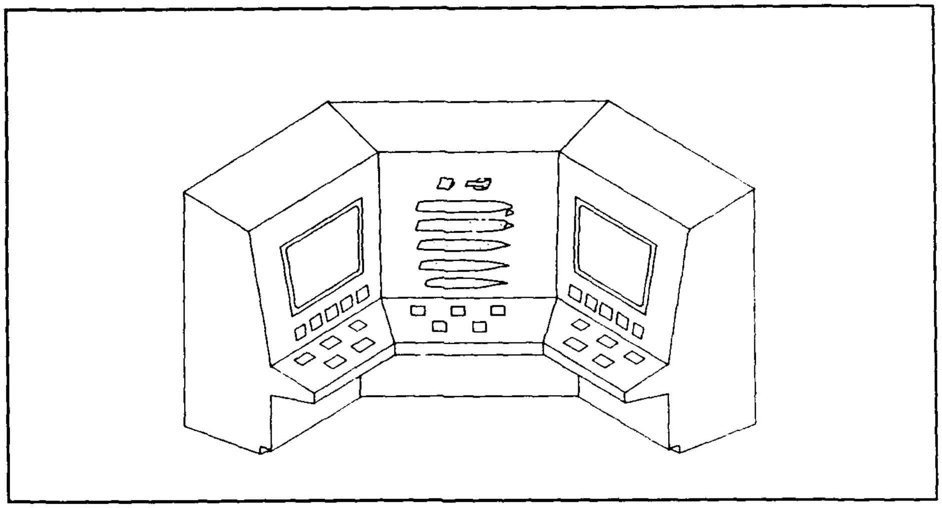


Figure 5.3 Conceptual Damage Control Console

5.4 Validation and Testing

This phase of the project focuses on determining whether the system, as constructed, actually solves the problem. There are several essential aspects of an expert system which must be validated. These include the method of reasoning, the validity of the conclusions, the rate of processing and the extent of coverage of the application.

This process is essentially equivalent to evaluating the input-output behavior of conventional software systems. The testing will consist of developing scenarios for the expert system, and letting it develop solutions. If the incorrect conclusion is reached, the issue will be similar to finding a software bug. A review of the rules used by the reasoning process should indentify the cause of the error. The expert system shells are quite useful for tracing the execution process.

In addition to evaluating the quality of the solutions provided, an assessment of the range and rate of solutions must be made.

5.5 Integration of the Expert System

The expert system can be integrated as an advisory, or as a control system.

5.5.1 Integration as an Advisory System

At this level of integration, the expert system monitors the sensor and controller status, and provides advice to the operator. A block diagram representation of this is shown on figure 5.5.1.

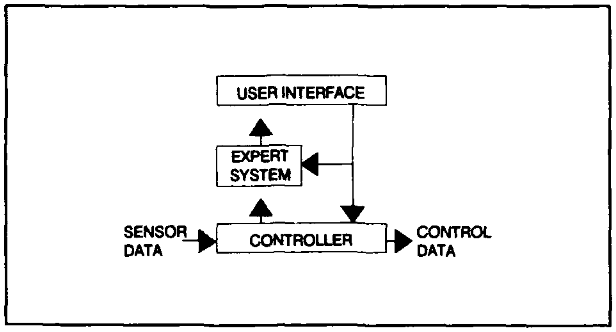


Figure 5.5.1 Expert Advisory System

5.5.2 Integration as a Control System

At this level of integration, the expert system monitors the sensor and controller status, provides advice to the operator and also develops the controller commands. A block diagram representation is shown on figure 5.5.2.

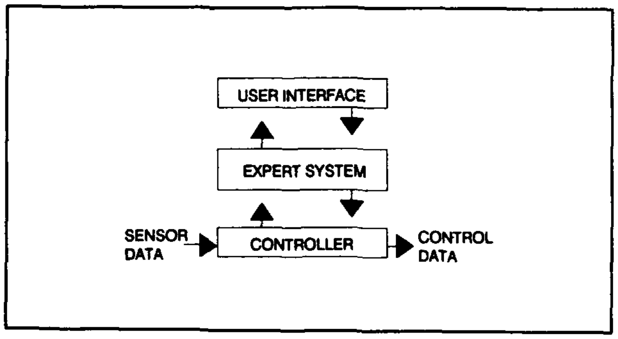


Figure 5.5.2 Expert Controller System

6.0 CONCLUSIONS

This paper has reviewed the state of present day damage control technology, and identified implementable means to achieve significant improvements in terms of system survivability and damage engagement capability.

The knowledge-based expert system technology offers a powerful capability for the damage control team to develop a real-time informed strategy for the containment and control of combat damage.

7.0 REFERENCES

(1) J. Kitfield, "How Survivable are Today's Warships?", Military Logistics Forum, March '87.

(2) "The Final Report of the USS STARK (FFG-31) Damage Assessment and Analysis", Washington, D.C., Naval Sea Systems Command, February 1988.

(3) "Lessons of the Falklands; Summary Report." Washington, D.C., Dept. of the Navy, February 1983.

(4) M. Williamson "Artificial Intelligence for Microcomputers", New York: Simon & Shuster, 1985.

(5) E. Tello, "Mastering AI Tools and Techniques", Indianapolis: H.W. Sams & Co., 1988.

(6) T. Felisky, G. Macheel, "Ships Transfer Data on Fiber-Optic Cables", Defense Electronics, March 1989.

(7) J. Walters, N. Nielsen, "Crafting Knowledge-Based Systems", New York: Wiley, 1988.

8.0 DISCLAIMER

The opinions expressed in this paper are those of the author and, as such, are not necessarily endorsed by the Department of National Defence (Canada).

DAMAGE COMMAND AND CONTROL
A PERSONAL VIEW OF FUTURE REQUIREMENTS

By R G Bryant BSc (Eng), C Eng, MIEE
Sea Systems Controllerate, BATH, UK

1. INTRODUCTION

The evolution of damage control and surveillance systems over the past thirty years has lagged far behind that for weapon and machinery control systems. In both these latter areas it is possible to identify separate of generations of system, with each generation using more sophisticated and automated means of controlling their respective equipments and with each generation adding extra functionality. In the field of damage control, however, there have been far fewer advances. Whilst modern electronics have been used in the latest systems the capabilities of many aspects of systems currently in use are not much greater than those provided in the early 1960s. For example, the use of manually marked-up stateboards, and heavy reliance on verbal reporting is still a feature of our present ships.

The reason for this is not obvious but I believe it is probably caused by a combination of these of these factors:

a. Damage control systems are insurance systems which, hopefully, will never be required in earnest. Such systems are frequently the first to suffer when cost ceilings are imposed and where the need is not demonstrated on a daily basis.

b. Machinery and weapon command and control systems are there to "serve" their respective end-user equipments. As the weapon and propulsion equipments have themselves become more sophisticated and complex there has been a corresponding need to improve the capability of the control system. This is not the case for damage command and control systems where the end user has been a man or men co-ordinating largely manual actions.

c. In the area of machinery control there are a number of civilian and industrial applications. These dominate the market and have led the field in developing products which naval customers have been able to modify. Industry also understands the machinery control problem fairly well and is able to contribute considerably to setting requirements and establishing how best to use the latest technology. In damage control, however, there is no similar industrial market or knowledge base and it is almost totally up to the customer to recognise where advancing technology can meet what previously may have been an unstated requirement. Whilst there is also no commercial market for weapon control systems the very considerable cost of the individual

weapon systems and the capabilities of the threat systems together impose a climate in which significant resources are demonstrably required.

During the 1980's, however, there have been a number of instances in which the need for more effective damage control has been made painfully apparent. The command, control and surveillance systems are among those in which improvements have been recognised to be necessary. Added to this are the extra capabilities required by the call to reduce the number of men in future ships. Since damage control is recognised as being a manpower intensive activity there is great pressure to use modern techniques to automate wherever possible.

2. DAMAGE CONTROL AND SURVEILLANCE HISTORICAL SUMMARY

Damage command and control in World War Two ships largely consisted of the Executive Officer rushing to the scene of the damage, coordinating action and reporting back to the captain on the bridge. Repair and containment of damage to machinery was the direct responsibility of the ship's engineer with little or no requirement to worry about incidents and consequences outside the engine room. Whilst the Executive Officer has retained his role of assessing the damage "on the spot" and keeping the Damage Control Officer briefed on the extent of the damage, the increasing complexity and interdependence of systems has demanded a more coordinated approach to damage control. A damage control headquarters (DCHQ) has therefore become necessary. In the early 60's this was in a convenient office and equipped only with stateboards and limited voice communications. However, in successive generations of ships the dependence of successful damage control on the ability to assess the state of and control the operation of a large number of ship's services has been recognised. The task of running DCHQ has therefore been given to the ships Marine Engineer Officer and the space itself has become more integrated with the machinery control room (MCR). In the latest frigate classes therefore DCHQ and the MCR have been combined into a Ship Control Centre (SCC) in which machinery and damage surveillance panels have been combined into a single console.

Alongside the evolution of the organisational aspects there has been a slow but real evolution in the facilities provided. These developments have been much slower than those in the weapon and machinery control fields for the reasons discussed earlier so the time and technology are now therefore right for dramatic improvements. Alarms and warnings for fire and flood have been provided since the 60's with a small increase in the coverage achieved and in the reliability of the sensors. The presentation of the information has been less than ideal with only the latest generation of systems making a serious attempt to improve the MMI. Voice communication facilities have been improved with primary and secondary links to fire and repair party posts, secondary damage control headquarters and the command positions. However, the compilation of a picture of the damage sustained is still reliant on men with chinagraph pencils marking up perspex display boards in each strategic position.

The latest frigate has provided for the first time a single damage surveillance system which senses fire and flood and monitors the status of important doors and hatches and gaseous firefighting systems. The information is presented on a large ship mimic which can also be used for the traditional marking-up of verbal reports. This panel forms part of the machinery control console in the SCC thus enabling the DC team to easily assimilate the status of other important systems. Also provided to the supervisors in the SCC is a computer based library of ship, compartment and system data which enables the operator to access information to assist in the decision making process and the directing of repair actions. The information provided is summarised in Table 1.

Compartment data:	Layouts Electrical isolation Air and fluid isolation Boundary cooling Flood and smoke removal Hazards
Shipwide data:	Electrical services to major systems Watch and quarter bill Route closure lists Gas drench guidance Jettison bill Tank capacities

TABLE 1 - DAMAGE CONTROL LIBRARY DATA

3. CURRENT REQUIREMENT

The successful application of damage command and control requires the sequence of events/activities discussed in Reference 1 and simplified in figure 1 to occur in an efficient and coordinated manner. When damage occurs it must first be sensed and the information transmitted to a person (or system) who has to decide on the appropriate action and then take the action (or order the action to be taken). Finally, assuming the sensor is still working, it will provide some feedback. The "INFORM" box at the centre indicates the need to report the current status to others with a need to know. Any or all of these activities can be carried out automatically or manually. Rapid reaction halon suppression systems and water spray systems are examples where all activities are achieved automatically. In contrast a man who discovers a small, smouldering fire and extinguishes it himself undertakes all the activities manually. Unfortunately, there is a need to design for much more major damage scenarios in which the simple examples given above can only play a small part. Each of the elements in Figure 1 will therefore be discussed to identify the requirements.

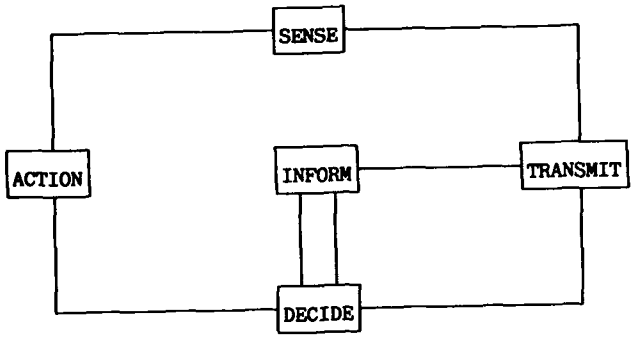


FIGURE 1 - DAMAGE CONTROL SEQUENCE

3.1 Sensing

Clearly there is a requirement for early detection of incidents such that their type, severity and location can all be reported as early and as accurately as possible. The traditional requirement to sense fire and flood will obviously continue but it is also necessary to ensure that all important systems have adequate instrumentation of their own so that any damage they sustain can be reported. It will also be necessary to monitor the closed down integrity of the ship. The degree of monitoring required for systems is to a level sufficient to enable system re-configuration to take place or so that the loss of capability can be assessed. If no re-configuration is possible or the system is not important then the DC personnel should not be burdened with excess data.

There is an argument that, since the damage likely to be caused by the worst case threat will destroy most of the sensors in the area affected, the depth of coverage need only be fairly limited. However this argument ignores the real threat from smaller incidents and peacetime accidents which are far more frequent and if allowed to develop undetected would soon become a major threat. Moreover, as most fires occur in harbour when fewer personnel are around to sense and act considerable reliance must be placed on the automatic detection systems. The coverage of fire and flood sensors therefore needs to be based principally on such incidents. It is not within the scope of this paper to discuss the requirements of each individual sensor type except to say that the number required means that they need to be reliable, relatively free of routine maintenance and accessible.

3.2 Transmission

Raw data from sensors needs to be transmitted as reliably as possible to those people or systems who have to act on that data. Where a system is acting automatically or processing of the data is required this should take place as close as possible to the source of the data. Since the damage control and surveillance is a shipwide system and DCHQ will need to initiate many actions a dual redundant means of bringing all data back is clearly necessary. Moreover, since DCHQ is itself liable to become untenable there is a need to provide the most important data directly to secondary DCHQ (HQ2).

The advent of independent zones within ships means it would also be desirable to provide the relevant data directly to a position within each zone, however, cost considerations would probably preclude this. The transmission of manually sensed information is also important. Traditionally this has been achieved using voice communications. These are notorious for allowing misinterpretation and confusion when a number of reports are being received against a background of high noise and stress. It is therefore essential to provide a means of making manual reports via data links. This will be discussed further in section 3.5 below.

3.3 Decision

The decision process is the most difficult to define and solve. Local automatic initiation of systems will not be considered further but rather the process which occurs in DCHQ and to a lesser extent at fire and repair party posts. The decision on what action to take, whether it be in response to battle damage at sea or to a harbour fire incident requires knowledge of a number of factors. The most important of these are shown in Table 2.

1.	Number, location and type of incidents
2.	Severity of and hazards involved in incidents
3.	Availability of defensive systems (firefighting, firemain, pumps)
4.	Availability of electrical power
5.	Availability, position and skill of manpower
6.	Command priorities
7.	Watertight integrity of ship
8.	Possible consequences/dependencies of actions

TABLE 2 - FACTORS AFFECTING DECISION

The man in charge has a potentially very difficult task and if the incident is in harbour or is un-expected he may not be highly trained and experienced. It is therefore necessary to provide accurate, easily read and understood displays of the nature and extent of the incident, of the status and availability of all support systems, of the priorities which the command wishes to place on particular functions and systems and of the watertight

integrity of the ship. He will also need access to a database of information concerning compartments, systems and the ship as summarised in Table 1.

There is a potential to swamp the operators with so much data that they cannot "see the wood for the trees". The display formats and content therefore need to be carefully designed and the operator given the option to select detail and information which he wishes. The custom of building up a large, manually compiled picture of all incidents on a single ship layout diagram using coloured chinagraph pencils has great merit in assisting the damage control officer to see the wider picture and should not be abandoned without a suitable replacement. There remains therefore the task for someone to compile a picture of the known, confirmed status of damage to be presented on a large scale display. The compilation process need not use chinagraph pencils but can be done electronically. In this way the picture is then available for transmission to other locations.

Given the complexity of the problem there is significant potential for the use of the emerging artificial intelligence technology to make the task easier. This could be used to automatically take action or merely to advise. Some of the areas where it is believed that assistance could be derived are listed in Table 3. Such a system would need to be given both sensor data and an encyclopaedic database and since to a large extent both these should already be provided in the SCC the basic building blocks are available. The degree to which AI techniques are required rather than more conventional software is, however, debateable whilst the technology is still maturing and establishing its most useful place in the programmers armoury.

Activities	Tasks
Firefighting Smoke removal Flood removal Stability calculation System re-configuration	Assessment of nature and location of incident. Prioritise and incidents Advise options to combat damage Advise on alterntive line-ups Advise on consequences of actions

TABLE 3 - ACTIVITIES AND TASKS WITH POTENTIAL FOR USE OF ARTIFICIAL INTELLIGENCE

3.4 Action

Damage control is traditionally a manpower intensive activity in which the adaptability, mobility and muscle power of men are used to combat the damage. Given the continuing pressure to reduce manpower there is a need to automate as many activities as possible. In this respect for firefighting the further application of gaseous fixed suppression systems and fixed water spray/fog systems will continue. In high risk compartments these may be automatically initiated but in others simple local and remote actuation is

all that is required. For flood removal a more extensive fit of fixed equipment will be necessary but portable manually "operated" pumps will remain essential.

The requirement to manually close a number of ventilation flaps and valves and to manually re-align the firemain and chilled water systems on receipt of alarm or going to the action state will need to be removed or drastically reduced. Such operations will need to be remotely actuated ideally by simple selection of the required mode. Similar consideration will have to be given to the remote actuation of door and hatch openings.

3.5 Informing

Up to this point the paper has concentrated on the need to provide detailed sensor information and displays in the SCC and in HQ2. A limited display of local data at each fire and repair party post has also been identified as an ideal. Alongside these detailed displays the requirement to provide a large scale picture of the recognised damage state has been discussed. This picture should be compiled in a single location, probably the SCC, and electrically transmitted to HQ2, FRPPs, the Operations Room, Weapon Section base and the bridge. Displays at these positions need not necessarily be on the same large scale as in the SCC but could perhaps be VDUs with an easily operated page selection system. The same links could also provide high level status information on important service systems. This information could come directly from the relevant machinery control system or could be manually inputted by the relevant system operator. The data links necessary between these stations would also provide a means for the operators to make manual reports to other locations. The data could be "broadcast" in a similar manner to the open line voice links currently used but avoid many of the drawbacks of using a verbal system. In this way it would be possible for the command to pass messages concerning priorities and weapon system requirements, to those co-ordinating the damage control activities. It would also be feasible to provide a number of connection points at strategic positions around the ship at which portable terminals could be used by action teams to report on the situation and/or seek information/instructions. Finally each operational terminal would need to have its own uninterruptable power supply with at least a 4 hour battery back-up. The data network would need to have a redundant path between each important operator position and if any bus controller function was necessary it would need to be duplicated.

4. TECHNOLOGY AND COST CONSIDERATIONS

Having considered in fairly general terms the requirements for fulfilling the functions identified in Figure 1, it is now relevant to discuss the technological trends/availability in each area and the cost considerations which might dominate the decisions on what is actually employed on future ships.

4.1 Sensors

The development of new sensors is being driven largely by the requirements of the process industry. Happily these are numerous and the range and capability of sensors are growing at a fairly fast rate. Happily, too there is a growing trend to standardise both the mechanical and electrical interfaces thus simplyfying the design of control and data collection units. It is not intended to discuss all sensors types relevant to damage control since they are numerous and most are perhaps more relevant to machinery control system design. However in the areas of fire and flood detection reduced manpower means that there is a need for each individual sensor to be addressed, for sensors to be capable of reporting analogue values (ie not just on/off sensors) and for sensors to be able to give a malfunction warning. Microelectronics are now making it possible and affordable to put the necessary "intelligence" inside the sensor or very close to it so that again controller design is simplified. It also means that data collection from a number of transducers can be achieved on a data transmission system without having to use complex units solely for data collection. Justifying the use of these more capable, and therefore more costly, sensors will have to be done on a through life cost basis. They will therefore need to live up to the claims of high reliability and/or lower preventative maintenance that are being made of them.

There is currently much discussion on the potential for using fibre optic sensors for many of the parameters required to be monitored. Currently, however, such sensors are not readily available and are expensive. It is therefore believed that it will be some time before they have established themselves as providing a cheaper or more reliable alternative to more conventional sensors. I do not expect them to be used in other than special applications until they are capable of being networked directly to a fibre optic data bus or until the fibre itself can be used as a reliable sensor in a shipboard environment.

4.2 Transmission

Data from a large number of sub-systems concerned with machinery, weapon and damage control will be generated and require transmission to a number of locations. Data transmission networks will therefore exist on the ship to provide for this. There are numerous debates on whether a single (albeit dual, triple or more redundant) highway should be provided to cater for all needs. *My own view is that this should not happen* - principally because the problems of managing the data flow requirements from a large number of users in a real control environment is probably too great for our procurement practices to handle but also because the traffic volume and system requirements would be dominated by weapon data with the potential for machinery and damage requirements to be considered secondary. However, the machinery control and surveillance and other ship service systems will become more capable on future ships and an Integrated Platform Management System is likely to be implemented (Reference 2). This system will have a shipwide data transmission network which will be resistant to damage and be carrying

data very similar in nature, origin and destination to that required for damage control and surveillance. This highway is therefore likely to be used to distribute damage control sensor data to the control positions required. The overall data rates, latencies and other communications requirements will be well within the capacity of network systems which are already available. Whilst there will be much debate on whether to use fibre optics or copper and whether a full OSI protocol stack is necessary and available, at the end of the day the system capable of meeting the requirements at minimum cost will be chosen. However, I also believe that since commercial applications are driving the development of capable fire detection systems which will operate with new analogue, addressable detectors cost and capability considerations mean that such systems with their own transmission networks will be used on a zone or area basis. In this way a fire detection controller in each zone or perhaps 2 or 3 for the whole ship will provide local display facilities but send the data to remote positions via the shipwide highway.

For the sub-system providing high level displays and allowing operators to send manually inputted data it is believed a separate data transmission system is required. Again technology is readily available to meet the requirement without any significant risk.

4.3 Decision

It is in this area that most development is required to provide the operator with assistance he needs. The provision of easily read displays, ergonomically designed controls and a coordinated approach to providing the correct data to each team member will require considerable investment in system engineering, human factors investigations and prototype designs. The techniques for achieving a good SCC design and user friendly operator displays and controls are now available and will need to be used to the full if our future ships are to be easy to operate. This will require considerable customer input over a prolonged period and will need a shared MOD/Industry approach to responsibility for the design.

Extensive use will be made of visual display units which will use windowing techniques, menu driven pages and panning and zooming capabilities to move around mimic diagrams. The information and controls available at the various positions will need to allow for both the peactime and action conditions which will inevitably have very different manning levels. The control consoles in the SCC are therefore likely to be multifunctional and the IPMS approach to platform system design will be essential if optimum performance is to be achieved. Whilst development costs for this approach, including provision of a shore based prototype, are likely to be considerable they are driven by the need to save manpower. If a separate damage control and surveillance system, covering fire and flood detection, firefighting, pumping and door/hatch status, were considered acceptable the development costs would be reduced although the extra redundancy and enhanced displays required would increase the UPC over that for current systems.

As discussed earlier the potential for use of artificial intelligence to assist the damage control officer is great and indeed some early studies into the firefighting application have been carried out (Reference 3). The technology, however, is still young and the cost overheads in procuring a very capable system are potentially very high. Considerable effort and commitment is required not only from the knowledge engineer but also from the expert. Unfortunately in damage control there appears to be many experts and almost as many solutions to any given problem. The other difficulty is that the proving or testing of a large AI system with a number of inputs and rules cannot yet be automated. Acceptance of a system therefore becomes very problematic especially if it was to be used for automated control purposes. Development of a large AI system would therefore require a number of DC "experts" to be available for a prolonged period and currently it is not yet possible to justify this investment.

Whilst it is feasible to implement a small AI system which carries out only advisory functions there is a danger that in simplifying the problem to make the system affordable it becomes one that does not need AI techniques to solve. It is therefore considered that a step by step approach to the use of AI languages in damage control systems will be employed. The use of a large rule base to make decisions on a large sensor input will not come quickly. Instead the first applications will use relatively small sections of AI code which is perhaps more efficient in tracking down information in a database or solving a few simple rules than is a more convential language. I also believe that the windowing display techniques and ability to build up a relational database (or knowledge base) which are features of most AI environments will be used to a large extent in the next generation of systems.

This leads on to the provision in the SCC, in HQ2 and to a lesser extent in the weapon section base of an encyclopaedic knowledge based system as discussed in section 3.3. The technique and usefulness for this has already been demonstrated in current ships and the only difficulty in expanding the system is the cost of data capture and data "upkeep". This problem may be solved in the future as the ship design and build process begins to use more and more CAD/CAM facilities which mean that most of the data should already exist in a computer database; the only problem will be to put it into the required format. The system provided should allow, even largely untrained personnel, easy access to the data and should have battery backed supplies. The database at each location should be independent and given the cheapness of modern proprietary hardware it is believed there should be at least one portable system on board. Assuming that a briefing system of the type described in the next section is employed then this database could be implemented on the same terminal hardware.

A further requirement in the SCC which has been much debated over the past few years is the provision of a stability monitoring or assessment system. Risking the danger of being contradicted by naval architects I believe that computer technology and flood level sensing equipment has now reached the state whereby the DC supervisor can be given a facility which

will perform calculations on the curent stability situation. This may well require the supervisor to manually enter some data - or at least confirm what has been automatically sensed - and it may only give a first order approximation - but it does not seem correct that we should continue to ask the operators to use calculators, pencils and paper with only a few worked examples (locked up in a suitably secure cupboard) as guidance. In the highly stressful situation they would be in when stability is a real concern then quick, reliable access to some guidance on status is almost essential.

4.4 Action

The past few years have seen the emergence of automatic rapid reaction spray systems and explosive suppression systems and an increased use of fixed manually operated water, foam and halon systems. Future ships will continue this trend and probably make fairly widespread use of automatic rapid reaction gaseous suppression systems, in an attempt to extinguish fires at source. Further discussion of the relative merits of firefighting systems will not be pursued here but again the technology required is largely available. The cost of such systems is therefore likely to dominate decisions on the extent of coverage.

Automatic or remote control of system valves, vent flaps and doors/hatches is an area where increased reliability and lower costs of actuators will be required if full use is to be made of control system potential. The reduced manpower requirement will undoubtedly drive the fitting of a number of such actuators but will also mean that maintenance labour will not be available to cope with an increased load. Closer attention will therefore need to be paid to system design to minimise the number of valves required whilst maintaining the ability to re-configure and partition systems. Key valves and flaps will then need to be automated.

Whilst many operators would perhaps like to see the capability to close all important doors and hatches at the touch of a single button (in the Operations Room) when a threat is imminent I do not believe that this will prove cost effective given the numbers involved in current ship design. Moreover if the number of men in the crew reduces to a level which makes this too large a task to handle then the requirement for the current numbers of doors and hatches to be left open to start with should reduce drastically. Remote actuation of doors and hatches is therefore likely to be restricted to a few, often used doors on the main communication deck.

4.5 Informing

As stated in section 3.5 it is believed there is a need for a data link between key positions to allow manually inputted data to be sent and to allow the recognised damage status to be available to all parties. Technology for achieving this is again available and indeed trials on a prototype system have been carried out. The display formats, especially those requiring a whole ship picture, need to be carefully designed but this is not seen as being difficult. The means of entering the data into the system also needs

full consideration as the use of a conventional keyboard will not be satisfactory. The choice would appear to lie between the use of a trackerball with special purpose keypad and/or softkeys or a CAD digitiser tablet with a special overlay. Neither option presents a difficult technological challenge except perhaps in the provision of a portable facility.

Future ships will almost certainly have a requirement to send a variety of other slowing changing information pictures to key positions around the ship. It is therefore considered likely that these functions could be combined onto a single system. The requirement for redundancy and battery backing of the DC element should, however, remain as a mandatory feature. The cost of provision of this network which would fulfill a number of requirements outside the damage control area will be much less than if a separate approach is taken to both problems and no real difficulties can be seen in its adoption.

The large scale whole ship display required in the SCC is an area where further technological developments are thought to be necessary before it can be fully realised. The display needs to consume fairly low power levels, be readible in all light conditions be large enough to be read from at least 4 metres and have a colour capability - as well as meet the usual naval environmental standards. It is not believed that this can currently be met but developments in colour LCD technology are promising.

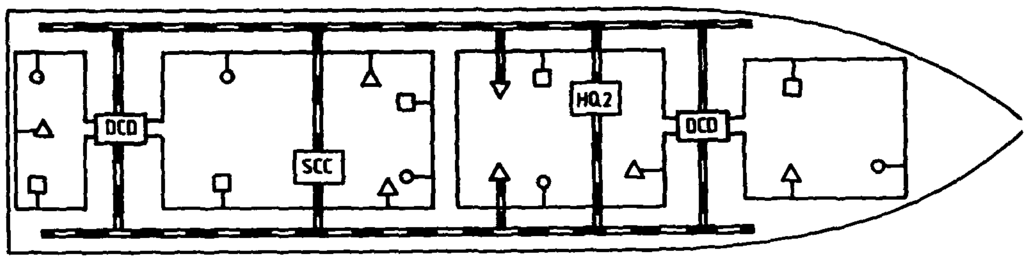
5. FUTURE SYSTEMS

Future damage command and control, systems will make far greater use of information technology driven both by the increased functionality which has been shown to be necessary and by the fewer number of men who will be available to carry out manual operations.

I believe that the next generation systems will need to provide a surveillance system which will give more extensive and accurate monitoring of fire and flood and a full monitoring of door and hatch status. Display of this information in the SCC will need to be integrated into those for essential shipwide services. The data will need to be available in a secondary damage control headquarters and will need to have a redundant data transmission path. A possible system architecture is shown in Figure 2.

Displays in the SCC will largely be on VDU's using mimic diagrams and windowing and paging techniques. Multifunction consoles designed to optimise peacetime and action manning levels will be used with the DC function coming under the umbrella of an Integrated Platform Management System.

Alongside this system will be a reporting and briefing system to enable operators to make damage reports, to allow the command to input priorities and to provide to all users a common picture of the damage status. The system will also provide key users access to a shipwide database to allow informed decisions to be reached. The facility will also be used to provide



DCD - DATA COLLECTION AND DISPLAY UNIT (SITED AT FIRE AND REPAIR PARTY POSTS)

△ ○ □ - FIRE, FLOOD, DOOR ETC SENSORS

- SHIPWIDE DATA HIGHWAY (DUAL REDUNDANT)

- LOCAL DATA COLLECTION NETWORK

- CONNECTIONS TO OTHER SYSTEMS (e.g. MCAS, COMBAT)

SCC - SHIP CONTROL CENTRE DAMAGE CONTROL DISPLAY SYSTEM

HQ2 - SECONDARY DAMAGE CONTROL HQ DISPLAY

FIG 2 - POSSIBLE DAMAGE SURVEILLANCE AND CONTROL SYSTEM ARCHITECTURE

a number of other information pages to be inputted on machinery status and enable other pages relevant to the command and weapon operators to be displayed. Key positions will be linked by dual redundant data highways and will have battery backed power supplies. A possible layout for this system is shown in Figure 3.

This system is only likely to be used for damage control purposes in action conditions but for normal peacetime cruising would provide a valuable means for operators with a need to know, to have access to data on those systems in which they have an interest but which they do not directly control.

6. SUMMARY

Overall it is believed that the basic building blocks for providing a more capable damage command and control system for the next generation of ships are already available. The difficult job is to specify the requirements properly and then to integrate the DC requirements with those of other systems to ensure that the ship can be operated as a whole in both peacetime and action conditions. This will not be an easy task and will require a fuller analysis of the requirements than has been possible in writing this paper. Developments already made and those envisaged for the next generation will extend the integration with weapon and machinery systems, will start to use AI techniques to assist the operators and will increase the automation of a number of systems.

References:

1. D J Clements and G P Kneebone, "Integrated Damage Surveillance and Control in Surface Warships", Inst Marine Engineers, Conference on Ship Fires in the 1980's, December 1985.

2. M I Hawken Future Direction for Royal Navy Machinery Control and Surveillance Systems. Ninth Ship Control Systems Symposium, Washington DC. September 1990.

3. J A Glen Knowledge Based Systems within Damage Surveillance and Control. Ninth Ship Control Systems Symposium, Washington DC. September 1990.

All views expressed are those of the author and do not necessarily represent those of HM Government.

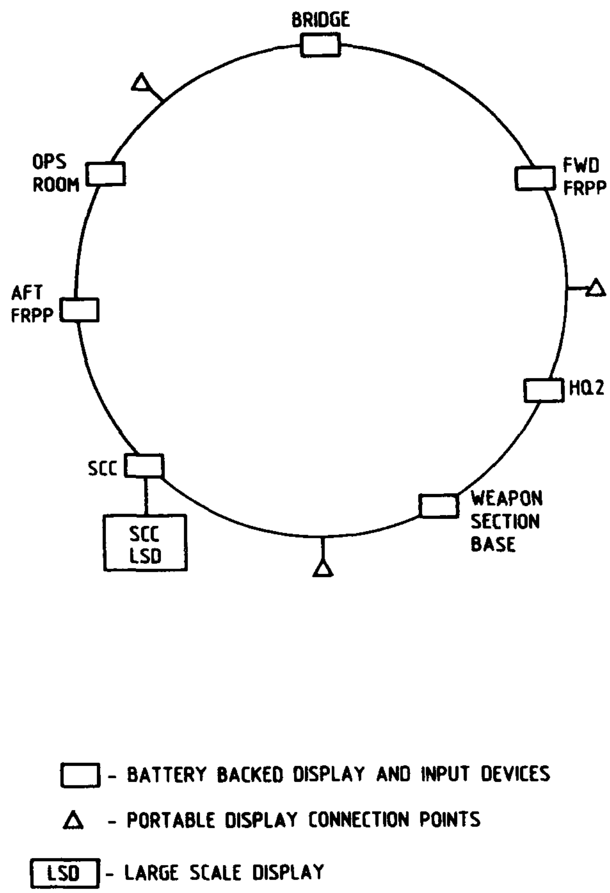


FIG 3 - DAMAGE REPORTING/BRIEFING LINKS, POSSIBLE SCHEMATIC

PC COMPATIBLE MODELING TECHNIQUES FOR INTER-CONSOLE COMMUNICATIONS

by CDR Fred Wyse, U.S. Navy,
Michael V. Post, John P. Mazurana, and Frank A. Lijewski
Westinghouse Electric Co. Machinery Technology Division

1. ABSTRACT

This paper presents an in-depth analysis of a modeling technique which can provide critical design information on expected system performance, error rates and capacity utilization; and can isolate system bottlenecks prior to a project's prototype development stage. It involves a Monte Carlo type simulation program which utilizes an event driven kernel routine. Desirable model input parameters and the types of output and statistical figures of merit which are generally most useful are discussed. In addition, specific problems and solutions are discussed which were encountered while using a Pascal based implementation of this modeling technique to analyze a communications network being installed in Machinery Control Systems (MCS) aboard DDG 51 Class ships. Information is also provided on other modeling techniques and other available software packages.

2. INTRODUCTION

This paper has been written in order to bring attention to some important consequences of using multiple embedded processors in complex control systems.

Digital communications have begun to play a major role in machinery control system designs. The simple point-to-point inter-unit communications systems of the past are being supplanted by local area networks. The resulting increases in system complexity are frequently accompanied by a myriad of problems which have not been previously experienced. Techniques for sizing communications capabilities, such as calculating system loading factors, are inadequate to fully predict the performance of today's communications networks.

In the case of the DDG 51 Machinery Control System, it was stipulated at the onset of the design phase that the system would use six embedded AN/UYK-44 computers. In order to analyze the communications problems subsequently encountered, a computer model of the process was deemed necessary.

Simulation models of system prototypes offer a number of advantages (1). Most important is they provide the ability to assess alternate design approaches. A complete set of performance measures can be established and evaluated on a

comparative basis. Simulation models also allow the analyst the opportunity to embed the desired level of realism into the analysis. In most cases, the models are simple and flexible. Most important, the simulation process is cost effective since designs can be verified before implementation, thus eliminating false starts.

While other system analysis techniques are available, such as analytical models based on sets of equations, these approaches require simplifying assumptions. These assumptions are often too restrictive and unrealistic. Furthermore, solutions are restricted to the set of assumptions used in defining the equations.

Discrete event simulation describes a system in terms of scheduled events whose order of occurrence need not follow any simple equation or pattern. These events can change the state parameters within the system using logic unique to each event. Discrete simulation can also be defined in terms of a flow of these entities through a network.

Modeling of digital systems lends itself well to the discrete simulation modeling approach. For this reason, multi-processor systems and communications networks are frequently modeled using discrete simulation. General purpose simulation languages such as SLAM II have been successfully used in these applications (3).

In systems whose processes involve a significant number of logic-intensive steps, it is sometimes less time consuming to create a special purpose simulator program to model the particular system of interest.

This paper is written in two parts. Section 3 provides a general process by which a simulator program can be designed and implemented. Sections 4 through 6 provide specific problems, resolutions, and considerations which were found to be of significance during the DDG 51 program, and which are likely to be equally significant in one form or another in future designs of comparable complexity.

3. OPERATIONAL DESCRIPTION OF A GENERAL PURPOSE DISCRETE EVENT SIMULATOR

A discrete event simulator utilizes a chronologically ordered queue of 'events'. These events are records which contain such information as the name of the event; the time at which it will be performed; any required source, destination or path information; and, if a random access queue is utilized, a pointer to the next consecutive event. Each of these events corresponds to an individual subroutine which, when called, will cause a

particular set of changes to occur in the state variables being modeled.

This type of simulator uses a 'simulator kernel' routine which is called recursively until some end state is reached, or until operator action halts the program. This 'kernel' pulls the next event off of the top of the event queue and calls its corresponding subroutine. The subroutine then causes some pre-defined change to occur in the model's state variables, updates any desired statistics values and stipulates a set of subsequent events to be added to the event queue. The kernel routine then adds these new events into the event queue at the appropriate chronological locations and repeats the process again.

Figure 3.1 shows a block diagram of a basic simulation program as described above. The following paragraphs provide a more detailed description of the functions to be carried out by each block.

A. RUN INITIALIZATION

i) "INITIALIZE" block

This set of subroutines sets all state variables and configuration variables to their default values and sets all statistics variables to zero.

ii) "INPUT CHARACTERISTICS FOR EACH RUN" block

This set of subroutines provides a set of menus from which the operator can select desired model statistics, configurations, protocols and run times to be used during each simulation run. These menus can be customized to include modes which test proposed fixes to specific communications problems.

iii) "SET CONFIGURATION" block

This set of subroutines sets the state variables and configuration flags as required for the specific set of run characteristics selected from the input menus. It may also write the configuration to an output file if desired.

B. OUTER RUN LOOP (For Statistical Accumulation):

i) "RANDOMIZE STARTING STATE AND INITIALIZE EVENT QUEUE" block

This subroutine is performed at the beginning of each run. It generates a snapshot of the state variables at a random point in the process being modeled, and use this set of

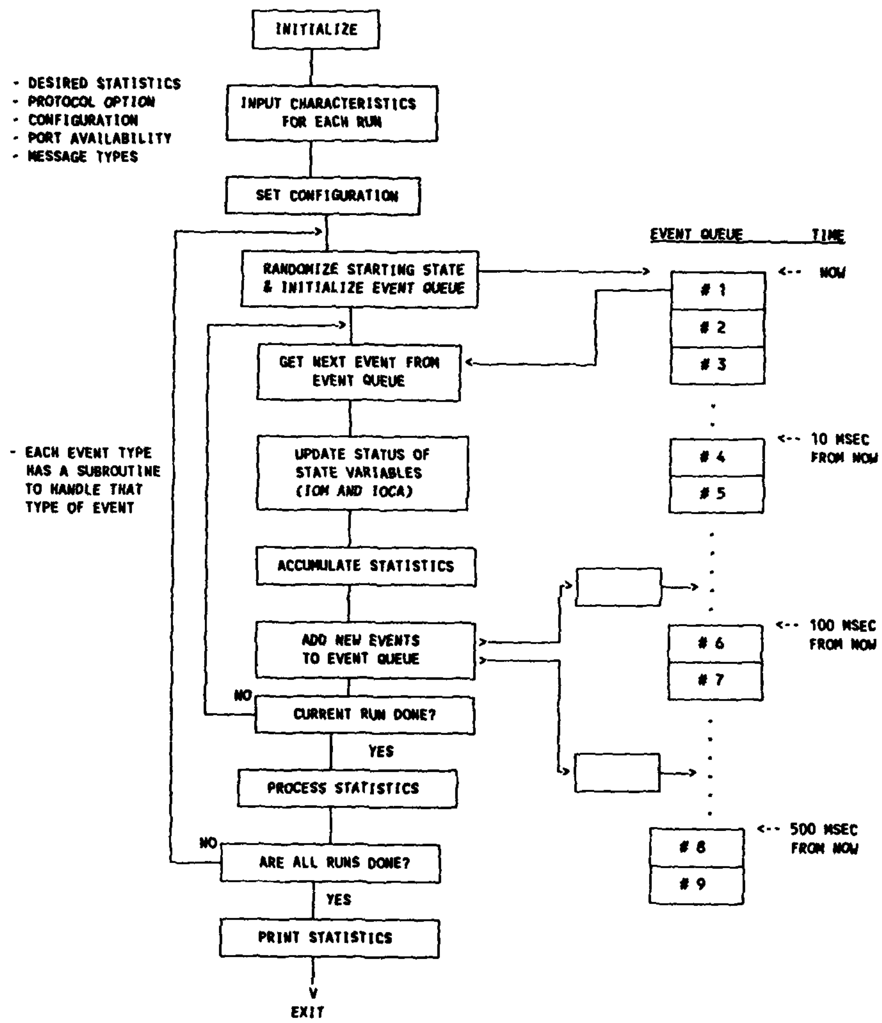


Figure 3.1
Simulation Program Block Diagram

state variables as a starting point for the simulation. It also places the necessary initial events into the event queue.

The 'Monte Carlo' simulation gets its name from this method of randomizing the starting point in the process being simulated for each of a large number of runs. Compilation of statistical data from each of these runs provides the basis for statistical analysis of the process. From this data, statistical distributions, averages and standard deviations can be determined for any of a variety of variables.

C. INNER SIMULATION KERNEL LOOP

i) "GET NEXT EVENT FROM EVENT QUEUE" block

This subroutine gets the next event record from the event queue, reads the data into a "current event" record, and deletes the event from the queue, moving the next subsequent event record up to the top.

ii) "UPDATE STATUS OF STATE VARIABLES" block

Each event has a corresponding subroutine associated with it. The subroutine associated with the "current event" is called at this point.

The event subroutines perform functions which change the model's state in some pre-defined way. They may also stipulate a set of new events which will subsequently be added to the event queue.

iii) "ACCUMULATE STATISTICS" block

Each event subroutine also increments a particular set of statistics variables. These variables provide the basis for statistical analysis of the process. Analysis of this data, following a set of runs, will allow statistical distributions, averages and standard deviations to be determined in a variety of areas.

If a degree of random variability in event starting times is to be modeled, then the Monte Carlo technique should be employed here as well.

iv) "ADD NEW EVENTS TO EVENT QUEUE" block

This subroutine adds the new events (stipulated in the previous routine) to the event queue at the appropriate location. This is done by comparing the required time of

occurrence for each new event with the times of occurrence required for each of the successive events already in the event queue.

D. END: (inner simulation kernel loop)

i) "PROCESS STATISTICS" block

This set of subroutines processes the statistical data compiled from each of the runs using various statistical analysis techniques. From this data, statistical distributions, averages and standard deviations are determined for all variables of interest.

E. END: (outer run loop)

i) "PRINT STATISTICS" block

This set of subroutines displays the results of the statistical analyses just performed. These results may be in the form of tables of averages and standard deviations, lists of relative process efficiencies, or even plots of statistical distributions if desired.

F. CONCLUDE RUN

At this point, the simulation program must close all files used during the simulation. It may also provide the operator with the option of re-starting the simulator and inputing a new set of run parameters.

4. ALTERNATIVE MODELING APPROACHES

To illustrate how the discrete event simulation process can be applied to digital system communication applications, the development and subsequent analysis of one particular communications system model is described. This particular model simulates the communications involved in the DDG 51 Machinery Control System. Sections 5 and 6 provide details on the DDG 51 MCS model and illustrate a successful application of the discrete modeling techniques discussed above.

Initial efforts focused on utilizing a general purpose package to develop a discrete event simulation model. The SLAM program by Pritsker Associates was evaluated for use in this application. Since well over twenty steps were required to represent a single message transfer, and a number of steps involved interaction with other messages or cancellation of scheduled events, the model quickly became cumbersome. Furthermore, the logic-intensive nature of the protocol at every step of the message transfer process complicated the model development process. Finally, even

minor changes to the logic in most cases required extensive re-work of the model. For these reasons, it was determined early-on, that a custom model would be more suitable for this effort.

The PC-based TurboPascal language was chosen due to its speed and versatile data structure features. High level languages such as Pascal can easily represent the complex protocol logic required; and the code, once written, can be easily modified.

5. THE MCS INTER-UNIT COMMUNICATIONS PROBLEM

The DDG 51 Machinery Control System handles monitoring and control of all ship propulsion, electrical, auxiliary, and damage control systems. It is composed of six computerized control consoles and a number of peripheral devices communicating over a common Data Multiplex System (DMS) Bus, as shown in Figure 5.1.

MCS operating stations, include the Propulsion and Auxiliary Control Console (PACC), the Engineering Officer of the Watch/Logging Unit (EOOW/LU), the Shaft Control Unit 1 (SCU1), the Shaft Control Unit 2 (SCU2), the Repair Station Console (RSC), and the Electric Plant/Damage Control Console (EPCC/DCC). Each console includes an AN/UYK-44 embedded processor. The processors exchange monitoring and control information over the DMS. In addition, a number of signals, such as damage control signals and the Bridge Control Unit (BCU) signals, are interfaced directly with the DMS.

The DMS consists of five coaxial cable bus lines which run the length of the ship. External signals are interfaced onto DMS at any of a number of Input Output Units (IOU), using a variety of interfaces. The MCS computers interface to DMS using the NATO STANAG-4156 serial interface. Each of the computers utilizes two of these 4156 channels (A and B) to transfer messages to and from the DMS. The IOUs are in turn interfaced to Remote Multiplexers (RM) which contain instructions for constructing and forwarding DMS message packets to destination IOUs. The RMs are interfaced to Area Multiplexers, which in turn interface with Traffic Controllers that control access to the five main DMS busses.

Since the DMS provides a total overall 24 MBit/Sec bit rate, and the MCS application requires a small portion of this capability, inter-unit communications were not predicted to present any problem. However, during preliminary testing of the MCS, it was determined that approximately 20% of the message transfer attempts were failing. This was a substantially greater failure rate than the 0.005% failure rate originally predicted. An additional item of concern was the extent to which the system loading would increase at the onset of a system problem. Since

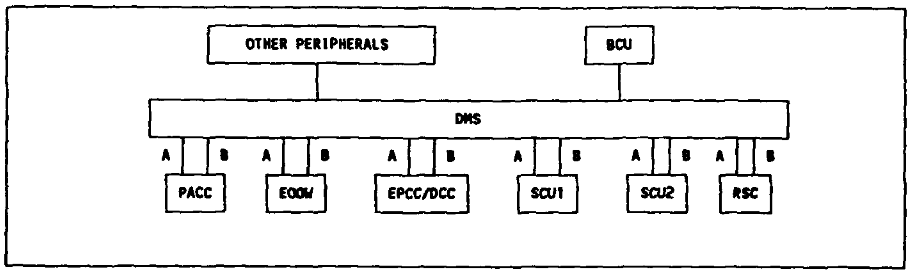


Figure 5.1
MCS INTER-UNIT COMMUNICATIONS

messages are retried up to three times if unsuccessfully transmitted, the system is further loaded, compounding the original problem. It is important as the limits of system capabilities are approached that the operator be provided with an indication of any impending communications problems prior to the system performance actually becoming degraded.

As a result, a DDG 51 MCS communications system computer model was proposed as a means of testing potential system changes which were being proposed to alleviate specific problems.

The objective was to construct a model which would operate on a time scale within one order of magnitude of real time, and which would predict results sufficiently close to those actually observed so that performance trends could be reliably extrapolated for any system modifications proposed. It was reasoned that such a model could save the Navy and its contractors a substantial amount of time and money by allowing quick analysis of effects of proposed system changes without requiring costly hardware and software changes. The proposals producing the best results in the model could then be implemented without risk of inefficient or unexpected system response.

The model could also be used to predict system efficiency in various casualty conditions, and to troubleshoot future system problems should they arise.

The model allows the operator to input various system configurations and modes of operation; and returns statistical data showing average relative percentages for path usage; message failures; successful message transmissions on first, second, third, and fourth try; port busy level; message collisions; channel failures; and timeouts.

The model is capable of operating on any IBM compatible personal computer when provided with a compiled version of the model program or with the applicable programming language and source code for the model. Modes of operation include Transmit Request protocol and Receive Request protocol, various levels of system degradation, and test modes capable of modeling each of the communications system changes proposed as performance enhancements.

Validation of model accuracy and reliability was accomplished by comparison of modeled system responses with those actually observed on the real system under similar circumstances.

Development of the model brought attention to a potential problem area not originally considered. That is, the sensitivity of system performance to various levels of degraded operation, as may exist on board a ship during casualty situations. The model proved to be of great value in this area, by allowing various system improvements to be tested in these degraded modes of operation.

These validation of model accuracy was expanded to include a broad range of system configurations, including several levels of degraded operations in which from one to six communications ports were disconnected to simulate component failures.

Figure 5.2 summarizes the message transfer process between two processors on the DMS, using a Transmit Request Protocol. MCS processor to processor communications involves four elements: source processor Input/Output Channel Adaptor (IOCA), DMS source Input Output Module (IOM), sink IOM, and sink IOCA. While there are a number of intermediate DMS process elements which are part of the message transmission process, these are not included in the model. The wide availability of DMS bandwidth permits this simplification.

TR transfer involves four steps. First, an outgoing message transfer is requested, a message path must be reserved between the source and sink IOM. The source IOCA forwards a two word transmit request command to the sink IOM. This creates a reserved path between the source and sink IOM. When the sink IOM has been reserved, the source IOCA is notified using STANAG 4156 interface control signals. The message is then transmitted by the source IOCA to sink IOCA. The third and fourth steps repeat the same sequence; however this time from message sink to message source. These two steps provide acknowledgement notification of message receipt.

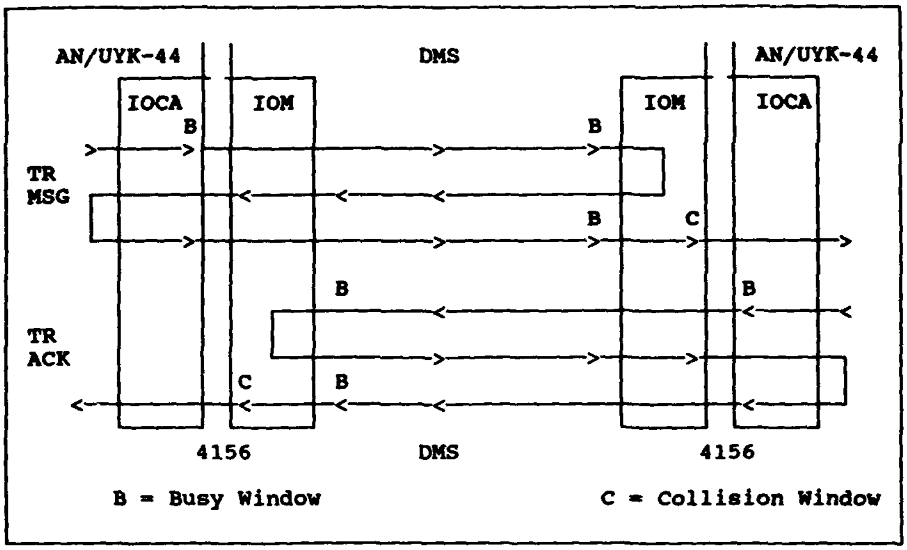


Figure 5.2
TRANSMIT REQUEST PROTOCOL

The TR protocol does require handling situations in which the sink or source is busy either transmitting or receiving another message. Also, the protocol does have short time intervals in the transfer process, during which a message collision occurs. These collisions result from not forwarding TR commands all the way to the IOCA.

A number of protocol improvements were suggested by various parties. Some of the changes would require significant redesign of the communications handling software. In particular, a simplified Receive Request (RR) Protocol, shown in Figure 5.3 was proposed. Other changes while relatively simple to implement, would still require relatively complex and time consuming measurements to evaluate the effectiveness of each change. It was determined that the simplest and most effective way to validate potential changes was to construct a simulation model of the communication process.

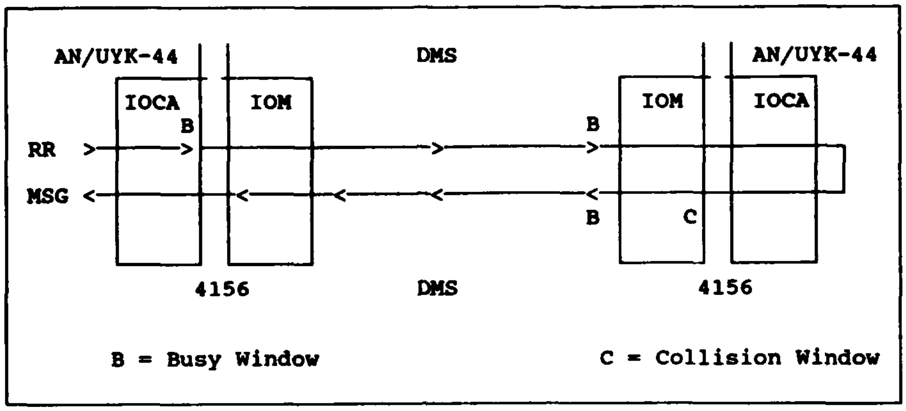


Figure 5.3
RECEIVE REQUEST PROTOCOL

6. PC-BASED PASCAL MODEL REPRESENTATION OF MCS COMMUNICATIONS

This simulation program developed for the DDG 51 was configured as a discrete event simulation model similar to the one discussed in section 3.

As described in section 5, the DDG 51 MCS communications system involves message transfers between source and sink processors, which can take place over any of four paths between the two source ports and two sink ports. Under the Transmit Request (TR) protocol, the messages transfers consist of two parts; transmission of the TR command word, which secures a path through DMS, and transmission of the actual message along that path.

Figure 5.2 summarizes the message transfer process by showing the order of communications port poling and reservation which takes place under the Transmit Request protocol.

The message transfer process is further broken down in the model, into a set of 22 distinct steps. Each of these steps constitutes and event which must be added to the event queue to be performed at the required time.

Figure 6.1 shows a timing diagram for the particular 11 events involved in completing a successful message transfer. The "Next Event" line shows the chain of new events added to the event queue in each step of the process. The times listed in this line

indicate the number of milliseconds between the events. These are the time increments used to chronologically place the events in the event queue.

The eight state variables of interest for the two applicable processors are shown under the "Port Reservations" section, with raised blocks indicating a 'reserved' status having been set during the appropriate events. "Collision Windows" are also delineated, during which an attempted communication from a third processor could cause both transmissions to be aborted.

Since each RM can handle up to two messages at once, dashed lines are used during RM reservations to indicate that the number of messages being handled by that particular RM has been incremented by one. The "T"s listed under specific events indicate the state variables which are tested during these events as part of the process logic used to determine what functions to perform next and which events to add to the event queue.

The following sections discuss some of the particular characteristics which were found to be most beneficial in the DDG 51 simulation model. They are organized a manner similar to that used in section 4, and the flow chart blocks referenced are those shown in Figure 3.1.

A. RUN INITIALIZATION

i) INITIALIZE

The model is constructed in modules to allow ease in altering and adding features as the need arises. It was found to be useful, during the debugging phase of the project, to add an event at the very end of the event queue with a very large time increment specified. The subroutine corresponding to this event would print out an error message indicating that the event queue was empty if the event were ever reached. During proper operation of the simulation, this event would never be performed since new events would be added to the event queue by at least one of the current events in order to perpetuate the communications process.

ii) INPUT CHARACTERISTICS FOR EACH RUN

It was found to be most efficient to provide several levels of option selection menus for the operator to use in selecting specific run characteristics. The initial menu, for instance, may allow run durations and frequently used default configurations to be selected, and may also allow for the selection of several different statistical output options, depending on the level of detail desired for a particular run. A subsequent 'customization menu' may then

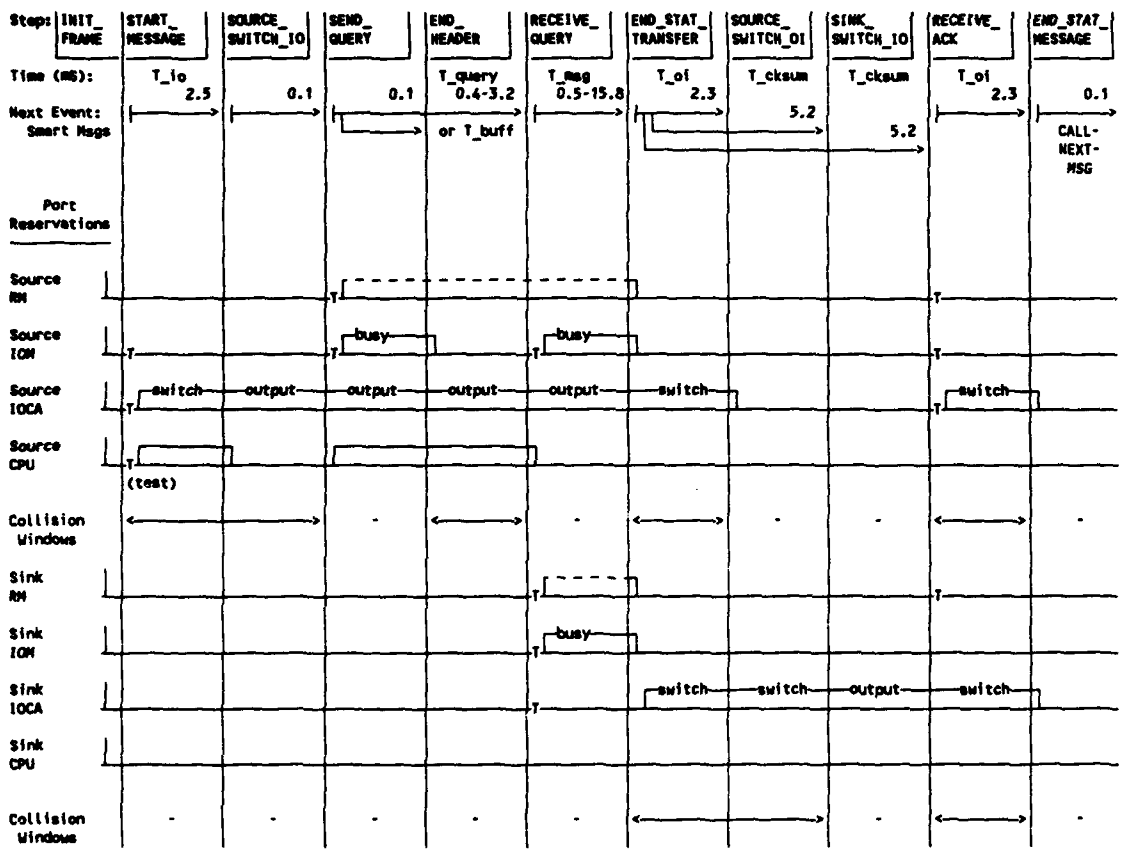


Figure 6.1
TRANSMIT REQUEST (Smart Messages only)

be called, if desired, to make detailed configuration changes.

Inputs to the model are made via selectable configuration menus on the terminal, and include duration of the run, DMS configuration for each port, port operability status, console operability status, communication protocol to be used, outputs to be produced, and combination of proposed system alterations to be tested. Outputs include average and standard deviation for message success rate on each attempt, message failure rate, channel failure rate, channel failure duration, timeout duration, and relative path usage; as well as various system parameter information useful in troubleshooting newly installed changes if selected. Rates are in events/hour or are a percentage of the total number of possible events during the run. All rates are calculated so as to be accurate to two decimal places.

iii) *SET CONFIGURATION*

It was found to be useful, for debugging purposes as well as data filing purposes, to write the detailed configuration to an output file, and to allow for this specific configuration to be re-entered in the input section if required for verification of correction of specific program problems.

B. OUTER RUN LOOP (For Statistical Accumulation):

i) RANDOMIZE STARTING STATE

Several randomization schemes were tried, and the standard deviations for all quantities were found to vary considerably depending on the scheme tried. Several of the schemes tried are listed below:

- randomized console starting times, once at the onset of each run (to test one random skew of transmission times)

- incorporation of a random amount of 'drift' between successive frame starting times for each console's message transmission time frame (to allow a slow 'walk' out of problem alignments of message transmission times)

- incorporation of a random periodic 'offset shift' of frame starting times for each console (to provide a means of periodically shuffling the relative message transmission times of all consoles)

- randomized console starting times, at every cycle through the kernel routine. (to average the performance of a large number of random skews of transmission times)

It was found that, if no dynamic randomization process was used during a run, each minute's worth of statistical data tended to be identical with the preceding minute's data, resulting in no standard deviations. The more degrees of randomization incorporated into the runs, the larger the standard deviations obtained.

It was also discovered, that the relative starting times of all console's message transmission time frames, had a very large effect on the overall average transmission efficiencies. The closer together the starting times were, the lower the efficiency of the system.

Unfortunately, the actual system, as understood, provided no mechanism for shifting the relative starting times of message transmission frames once the consoles were powered up. And the actual system did not experience the significant dependence on relative start up times predicted under these circumstances.

It was found to be of use to have an option in the main menu allowing selection of one of these randomization approaches, in order to easily rerun specific configurations using an alternate randomization scheme.

C. *INNER SIMULATION KERNEL LOOP*

It was found to be very convenient to be able to halt the model at any time during its operation, and have the option to either continue the process from the point at which it was stopped, alter the subsequent run configurations and continue, or end the simulation entirely and print out the statistical results up to that point. This was accomplished by including a check of keyboard input at the beginning of each cycle through the simulator kernel loop.

i) GET NEXT EVENT FROM EVENT QUEUE

The event queue was initially configured as a sequential array. Each time an event was added, all events in the queue with times greater than that of the event being added were shifted back one position. Conversion of this array to a 'random access' array using pointers to subsequent events was considered. A measure of maximum number of events in the event queue typically indicated about 20. Since most of the new events are added in the bottom half of the queue, the number of events being shifted was fairly small, and the gain in converting the array was considered to be minimal.

ii) UPDATE STATUS OF STATE VARIABLES

During the troubleshooting phase of the project, a dedicated troubleshooting routine was added to the program and repeatedly proved to be of immense value when adding new options to the model. This troubleshooting routine consisted of a single subroutine which tested each of a large number of variables of interest each time it was called, and compared each with the value recorded during the previous call. For each value that had changed, an entry was added to a troubleshooting file along with the name of the subroutine in which the change occurred.

This technique of writing only changes to a file enabled a large number of variables to be tested at a large number of points in the program, while still avoiding a massive accumulation of data in the file. In order to minimize any reduction in program speed when the option was not in use, the calls to the subroutine were incorporated into 'IF' statements, and only performed if a 'TS' troubleshooting flag was set. An option was added to the main menu enabling this flag to be toggled on and off as desired.

A statement of the form:

```
IF TS = On THEN Trouble_Shoot('xxx')
```

was added to the end of each subroutine in the program so that any state variable changes made by that subroutine would be displayed in the trouble shooting file. The 'xxx' was replaced, in each case, with the name of the subroutine in which the call was made, and this name was included in the trouble shooting file along with a list of names and values for all variables which had been changed.

It was also found to beneficial during the troubleshooting phase to include 'Trip Point' statements in various subroutines which tested for conditions which should never occur (e.g. a 'Smart' message event being called by a 'Dumb' message which should never have reached that point). If any of these illegal conditions is found to have occurred, an error message can be printed or displayed along with the values of all applicable state variables at that point, so as to enable the problem to be easily determined during subsequent troubleshooting efforts.

iii) ACCUMULATE STATISTICS

Data is accumulated in global variable arrays, and results of statistical analysis are printed to files on disk for subsequent printing or viewing as desired.

It was found that the most useful statistics data consisted of average, minimum, and maximum values for most parameters. This data was easily obtained by maintaining a statistics record with counters for each of these parameters, and incrementing the appropriate counter each time a particular event routine was called.

At the end of each run, the totals of each of these counters was added to the corresponding field in a second record. The values of each field in this 'Summation' record would be divided by the number of runs at the end of the simulation in order to arrive at an average.

If standard deviations are also desired, than an additional record must be used to accumulate sums of the squares of each of these counter values after each run.

The 'Counter' record in then re-zeroed prior to commencing the next run.

iv) ADD NEW EVENTS TO EVENT QUEUE

At the conclusion of a message transfer (either successful or unsuccessful), an event must be added to the event queue corresponding to the commencement of a subsequent message; otherwise the process is halted after the first transmission attempt.

It was found that in some situations the particular 'retry' scheme used following message collisions could result in both messages being re-tried at times which would result in a repeat of the same collision. Care must be taken to construct the model accurately enough so as to avoid getting caught in this cycle.

D. END: (inner simulation kernel loop)

i) PROCESS STATISTICS

At the end of each run, the totals of each of the counter fields in the 'Counter' record are added to the corresponding fields in a 'Summation' record in order for averages to be calculated on completion of the runs. The squares of these values must also be added to corresponding fields in a 'Sum_of_Squares' record if standard deviations are also to be calculated.

The 'Counter' record must be re-zeroed prior to commencing the next run.

Another set of statistics which was found to be of use involved averages and standard deviations for an entire set of variables (as for the 'Total' number of message failures for all consoles, as opposed to number of message failures for each individual pair of consoles).

In order to determine these averages, the appropriate fields in the 'Summation' record could simply be summed and divided by the number of runs. In order to calculate standard deviations for these totals, however, an entirely new statistical record was required to accumulate sums of squares of the total counts of all applicable counters during each run. If the number of variables being monitored is very large, the size of the statistics records will be very large, and the memory requirements will grow very rapidly as the records are added.

In the case of this particular model, the size of the program and memory requirements grew to the point that the TurboPascal could no longer compile the program directly into memory. The program could still be compiled, however, by setting the TurboPascal compile destination to 'Disk' (under the TurboPascal Main/Compile menu), and then setting the TurboPascal linker destination to 'Disk' (under the TurboPascal Main/Options/Compilation-Options menu), then rebuilding the program and running it.

One benefit derived from the model's statistical output, was that it helped to show the correlation between the frequency of diagnostic messages displayed (channel timeouts) and the actual system operating efficiency (number of messages lost).

E. END: (outer run loop)

i) *PRINT STATISTICS*

It was found to be helpful to allow for several levels of output detail to be selected. The most universally useful output tables contained statistical averages, standard deviations, maximum and minimum values for each parameter on a console by console basis as well as on a total (of all consoles) basis.

Also of use during development and verification of all simulation options and subsequent modifications, were lists of message transmissions on a frame by frame level; as well as lists of events and state variable changes on an event by event level.

The troubleshooting routine discussed earlier was also of immense value, and was designed so as to write supplemental state variable information into the events level summary file.

F. CONCLUDE RUN

All files opened over the course of the simulation must be closed upon its completion. A final message to the user could also be printed on the screen giving instructions on how to view the output files of interest.

7. <u>CONCLUSION</u>

As a result of the experience gained in developing the DDG 51 MCS communication system, several important lessons have been learned. In particular, implementing complex control systems using multiple embedded processors over local area networks can result in unexpected problems, not revealed by simple analytical techniques. Simulation models can be quite useful in predicting communication system performance. This is particularly true if a non-standard protocol is used, as was the case in the DDG 51 MCS. These models can also enable more trade-off studies to be performed during the design phase. This allows the potential changes in system performance due to alteration of various system parameters to be more accurately predicted. Developing a simulation model also forces the designers to review the technical details of the interfacing hardware and associated software protocols, where subtle differences can significantly impact performance. Finally, valuable system integration time can be saved when the modeling effort results in a smooth implementation.

REFERENCES

(1) M. Al-Holou, J. B. Ghosh, D. B. Rogers, "General SImulation of Multiprocessor Interconnection Networks", Proceedings of the 1985 Winter SImulation Conference, D. Gantz, G. Blais, S. Solomon (eds.), page 352-357.

(2) J. L. Arthur, "Microcomputer Simulation Systems", Computing and Operations Research, Pergamon Journals LTD., Vol. 13 No. 2/3, 1986, page 167-183.

(3) Y. Chang, and S. Shen, "Simulation Investigation on Message-Based CSMA/CD Priority Protocols", Simulation, May 1988, pages 203-214.

NON-INTRUSIVE MACHINERY MONITORING AND DIAGNOSTIC SYSTEMS FOR SURFACE SHIPS

V. B. (Kisan) Pandit
Peter M. Grotsky
Naval Sea Systems Command
Washington, D.C.

Henry Gruner
Dundics Enterprises, Inc.
U. S. A.

The views expressed herein are the personal opinions of the authors and are not necessarily the official views of the Department of Defense or the Department of the Navy.

NON-INTRUSIVE MACHINERY MONITORING AND DIAGNOSTIC SYSTEMS FOR SURFACE SHIPS

V. B. (Kisan) Pandit (NAVSEA SEA 915/935)
Peter M. Grotsky (NAVSEA SEA 56X3B)
Henry Gruner (Dundics Enterprises,Inc. U.S.A)

1.0 INTRODUCTION

The United States surface navy today faces a significant dilemma, which can best be described as one of how to operate and maintain the fleet in a high state of operational readiness with fewer dollars and fewer people. It has long been recognized that the navy needs to do away with insurance based and time-directed repair philosophy, and institutionalize, the concept of condition- directed repair to minimize unnecessary repairs and be able to prioritize repair decisions based on sound engineering assessment. Many of the equipment and system failures that impact operational readiness of the surface ships can be prevented if the operator could monitor the performance degradation and accurately access the material condition on-line with the use of reliable expert monitoring and diagnostic systems.

Naval Sea Systems Command's (NAVSEA) Surface Ship Maintenance Division (SSMD) was established as the central point of contact for NAVSEA-TYPE COMMANDER (TYCOM) partnership in formulating surface ship maintenance policies and to provide an organization and resources for execution of maintenance systems assessment, development, and improvement efforts. SSMD has been at the forefront in introducing state of the art technology to improve maintenance planning and material condition assessment. This paper will discuss SSMDs efforts in conjunction with NAVSEA technical codes, particularly the Internal Combustion and Gas Turbine Engines Division, in introducing and demonstrating the use of commercially available, reliable, off-the-shelf technology for non-intrusive machinery monitoring and condition assessment. The focus has been to concentrate on technology that will enable ship operators to assess the condition of equipment on-line and to provide expert diagnostic tools that will assist them in determining an appropriate level of preventive and/or corrective maintenance tasks, in a manner to prevent catastrophic failures. The emphasis has been on low cost reliable technology that will provide Return on Investment (ROI) in the shortest period, improve

ship board safety and platform material readiness.

The systems that will be discussed in detail are Diesel Engine Monitoring and Analysis (DEMA), Automated Diesel Engine Trend Analysis (ADETA) and Fireroom Maintenance Management Support System (FMMSS). DEMA has been installed on USS HARLAN COUNTY (LST-1179) since 1987 and FMMSS has been installed on USS T. C. HART (FF-1052) and USS WASP (LHD-1) since 1987 and 1990 respectively. ADETA was introduced to the fleet in 1989 and is currently being issued to all ships with diesel engines of 450 HP and above. The marriage of DEMA and ADETA will provide the navy with an on-line maintenance tool that operates automatically or on operator request for collection and analysis of maintenance data. This paper will further describe how DEMA integrated with ADETA allows real time analysis for predicting the need for corrective or preventive maintenance.

2.0 CONCEPT OF NON-INTRUSIVE SYSTEMS

Before we discuss the initiatives described above, the concept that has guided the development of these systems needs to be explained. The U.S. Navy has several active surface ships that have had installed integrated control and monitoring systems. As a rule, a control system on a Navy ship must meet rigorous reliability and performance standards. However, when a monitoring and diagnostic system that is to be predominantly used for maintenance planning and machinery condition assessment is integrated with a shipboard control system, by default, it must also comply with the same rigorous standards. The drawback being that the initial acquisition and the life cycle support cost of such a system skyrockets as compared to a similar system built to commercial standards.

As the cost escalates, the return on investment for such a system diminishes and it becomes increasingly difficult to justify the use of on-line monitoring and diagnostic system in todays budgetary environment.

To address the dichotomy of a need to have reliable and accurate tools of determining condition of shipboard machinery and to make such tools affordable, the Surface Ship Maintenance Division introduced the concept of "non-intrusive" systems. The two under lying principles of this concept are as follows:

i) The system should not interfere with the machinery it monitors. It should not be integrated with any shipboard control systems. A partial failure or a total failure of a non-intrusive system should not effect operation of the monitored machinery, nor should the system's functioning be essential for operation of

the shipboard machinery it is monitoring.

ii) A non-intrusive system by itself should not be perceived as a maintenance burden by ship operators. The system should be user-friendly, easy to operate and easy to troubleshoot.

By complying with these two basic principles, it is feasible to adapt commercially available technology for Navy use and at an affordable price.

3.0 PROPULSION PLANT CONDITION ASSESSMENT SYSTEM (PPCAS)

3.1 Background

The current PPCAS technology and initiatives for on-line, non-intrusive machinery monitoring and diagnostic systems in Navy surface ships can be traced back to the commercial marine industry.

As a result of a variety of economic factors over the last 10 years, the commercial marine industry has been facing major obstacles in achieving competitive levels of operations. Aggressive managers of both deep ocean as well as inland marine fleets have been actively searching for management tools and vessel operations related technology applications which would aid in maximizing revenues per ton-mile of cargo delivered, increase annual availability of their vessels at maximum capacity and reduce fleet management costs at headquarters.

In the inland marine industry, a series of applications evolved related to diesel engine monitoring and analysis technology for fuel management, monitoring of vital engine parameters, automatic logging, improved data transmission to shore computers, and port engineer/office analysis of operational data. The typical engine used by the towboats operating on the inland waterways is a marine diesel, 16 cylinders per engine, turbocharged with a horsepower output of 1000-3500 BHP. Each towboat is normally powered by two or three diesel engines. In addition, two diesel generators account for the electrical power requirements and other auxiliary systems such as lube oil pumps, transfer pumps, etc., support the vessel's daily functions. For a three engine 10,500 HP boat, approximately 140 engine room parameters are monitored. The evolutionary integration of these diesel monitoring and analysis functions on the ALCO 251 and EMD 645 diesel engines led to the development of a flexible, monitoring and diagnostic system architecture specifically oriented towards providing a complete, integrated repertoire of capabilities for performing on-line condition assessment.

3.2 PPCAS Design Principles and Objectives

Many guiding design principles and objectives which evolved in the inland marine applications paralleled objectives originally established by the Navy when PMS 306, the Navy Maintenance System Project Office, was in existence for Ship Integrated Maintenance Management Systems (SIMMS). This program was an integrated effort between the Chief of Naval Operations and the Naval Sea Systems Command. The objective was to provide integrated interfaces of shipboard monitoring systems with shipboard, intermediate, and depot level maintenance systems. The general design principles require that:

- Applications should be designed to support basic work and decision processes aboard ship and should not simply collect and computerize data.

- The computer should be employed to automate the preparation of maintenance-related paperwork to the maximum extent possible. Data that can be predetermined by the computer should be provided by the computer. Shipboard personnel should only be required to enter that information which they are uniquely qualified to provide.

- System hardware should be configured, and software should be designed, to provide an effective and efficient man-machine interface.

- The system should be designed for use by all rates and ratings and should not require addition of data processing ratings to ship's company.

- Training requirements should be minimized and the system itself should be self-teaching insofar as possible.

- The system should be designed to interface with all maintenance systems external to the ship (e.g., intermediate Maintenance Management Systems, Shipyard MIS, etc.) and with related shipboard systems, such as supply and personnel.

- Applications should be designed as an integrated system using an integrated data base. Data elements required by various portions of the system should be commonly acquired and managed, related modules should be designed to communicate intelligently with one another, and standard definitions should be used throughout the system.

- The design should ensure that maintenance managers receive information relevant to their decisions in a useful format.

o The system should include feedback mechanisms which are incentives for more accurate data reporting.

o System designers should establish need as the primary criterion for including data elements in the systems.

o The design should require regular reporting of the minimum data necessary to support maintenance decisions at all levels.

o Hardware/software design tradeoffs should consider system life cycle cost, not merely acquisition costs.

o Consideration must be given to expeditious transmittal (via telecommunications) of data to/from the ship.

Based on the above, there are certain elements of the program that could very easily be accomplished by a properly designed, continuous on-line machinery monitoring system providing necessary performance and condition monitoring, trend analysis and predictive maintenance information to the ship.

3.3 Navy PPCAS Application

The inception of PPCAS in the Navy can be traced back to two prototype installations sponsored by the SSMD and SEA 56X33 to demonstrate the applicability of on-line monitoring and diagnostic systems; namely, the Diesel Engine Monitoring and Analysis (DEMA) system and the Fireroom Maintenance Management Support System (FMMSS). The goal was to use commercially available technology in assessing condition of machinery on line, so that an operator could detect abnormal equipment operation and take corrective measures to avert catastrophic failures.

Both DEMA and FMMSS have been proven successful and have strong endorsements from the fleet commanders. Both DEMA and FMMSS use the same basic hardware and operating system. The application software in one case is tailored for a diesel plant and for the steam plant in the other case. The microprocessor and the hardware architecture of both systems is such that it has the capability to monitor propulsion plant auxiliary equipment as well as the main propulsion plant with minor system modifications. Such systems can be utilized to their full potential with marginal cost increases, thereby increasing their effectiveness and the benefits that the Navy can derive from their use.

During the evaluation phase of both prototype systems it was evident that technology has a much broader application and the system definition should not link it to a piece of equipment.

Therefore, it was decided to broaden the scope of both DEMA and FMMSS to include auxiliary machinery and rename it as the Propulsion Plant Condition Assessment System (PPCAS). PPCAS prototypes are currently being installed on USS WASP (LHD-1), USS AMERICA (CV 66), and USS GERMANTOWN (LSD 42). A Machinery Alteration (MACHALT) proposal to install PPCAS on LSD 41 Class has been approved. A system for gas turbine monitoring and diagnostic is also being developed by NAVSEA and is scheduled to be installed on USS CLIFTON SPRAGUE (FFG 16). With completion of this prototype system, PPCAS will have three distinct modules, one for diesel propulsion ships, one for steam propulsion ships, and one for gas turbine propulsion ships. The same central processing unit hardware, operating system, and maintenance management application software shell are planned for PPCAS propulsion and machinery applications such as diesel, steam, air conditioning, and distilling plants.

3.4 PPCAS Top Level System Design and Specifications

The PPCAS system is designed to provide complete machinery condition assessment, diagnostics, prognostics and maintenance management capabilities for a broad range of shipboard machinery. The applications software operates in a multitasking real time environment allowing simultaneous coexistence of foreground/background tasks. The PPCAS provides:

- real time data display of all available monitored parameters
- recall of performance deviation related recorded data
- recall and graphical representation of machinery trend data
- recall and graphical display of expert system based diagnostic advisories related to recommended maintenance actions.

To allow application of the PPCAS to a broad range of shipboard machinery without the need for software modifications, a complex application shell is provided for initial setup and on-line field modifications. All basic functions needed for defining and implementing an equipment availability management system are provided for in the system editor.

The system consists of a custom computer with a real time multitasking operating system and application software written specifically to perform shipboard machinery condition assessment and availability planning related functions. The components and enclosures have been selected to survive installation in the harsh environment found in marine propulsion/auxiliary spaces. Since the design was focused on the flexibility of application, it can be easily tailored to various ship classes without re-

programming or new application program generation. Figures 1 and 2 respectively show a typical PPCAS configuration and data flow diagram.

The system assesses performance and efficiency of machinery, as compared to the design baseline and relationship of the degradation to the operation of the engineering plant. This allows timely planning of the organizational, intermediate and depot level maintenance tasks through isolation of equipment degradations before major effects are realized.

The Top Level System Specifications are as below:

- Real time display of monitored parameters as they relate to expected performance
  - User can request information on any parameter, any subgroup of parameters or all parameters
  - User can design own displays with an editor to best view the data(tabular and graphical)

- Periodic log sheet utilities
  - Automated logging of all parameters onto user predesigned log forms
  - User can use an editor to enter any unmonitored parameters such as lube oil used, filter changes, etc.
  - Comment section for engineer
  - Printing of log forms on wide carriage color printer to allow highlighting of out of limit parameters

- Automated monitoring
  - Individual scan rates for monitored parameters including accelerometers and velocity probes
  - Performance comparison with the baseline curves or alarm values
  - Correlation between two or more parameters with .pa variable performance alarm ranges
  - Triggered scans, where further channels can be checked, data logged and alarms recorded
  - Real time expert based diagnostics, advisories and maintenance recommendations
  - Trending data scans triggered by individual parameters

- Database manager
  - Data transfer to a higher level shipboard and/or shore side computer
  - Flexible database language supported capabilities for data storage and display on demand of performance degradation triggered alarm files and trend files,

expert diagnostic files and maintenance management related logistic data file

- o Flexible system screen based editor
  - Complete sensor suite definition by channel
  - Individual channel scan rate establishment
  - Identification of scan groups of channels for performance logging
  - Event triggered scan sequence definition
  - Trend data scan group design and periodicity establishment
  - Data formatting for analysis
  - Log form data collection periodicity definition and formatting
  - Tabular and graphical real time performance data display formatting
  - Built in machinery function computing
  - Graphical expert system design input capability for trending diagnostics and maintenance advisories
  - Graphical input of expected equipment performance characteristics
  - Input of logistic management system elements

- o Performance baseline establishment
  - Easy graphical and tabular machinery/equipment performance map input facilities

- o Trend analysis/ predictive maintenance
  - Knowing baseline conditions of various components, the expert system based diagnostic module will monitor the health of the component continuously over a period of time and allow:

    a) automated scheduling of maintenance actions required in the future, based on analysis of performance degradation
    b) immediate diagnostic advisories

- o Vibration Analysis Features
  - On-line Fast Fourier Transform analysis for narrow band monitoring
  - Band variable alarm limits
  - Correlation with other process variables
  - Time based trending in conjunction with other variables
  - RPM tracking for variable speed equipment
  - Vibration profile and trend download to shore computer

- o Reciprocating engine analysis

- Cylinder cycle efficiency analysis and graphical display and trending
- Pressure/volume and pressure/crank angle performance assessment trending and display
- parameter performance data storage by machine and cylinder for time based comparative analysis

o Maintenance support software
- Computerized daily, weekly and monthly schedules of activities for engineer supported by expert system based condition analysis of the monitored systems integrated with time directed planned maintenance
- Interactive mode allows maintenance engineer to enter work performed, conditions found and material used for historical analysis
- Expert system based diagnostics provides rapid fault identification and corrective action recommendation

3.5 Steam Propulsion System Application - Fireroom Maintenance Management Support System (FMMSS)

In steam driven ship propulsion systems, oil fired burners furnish variable and controlled amounts of steam to the propulsion system through a steam throttle. Typically, the boilers are controlled automatically by a pneumatic control system, which responds to the steam pressure in the boiler drum, the rate of steam flow through the steam throttle, the rate of feedwater flow to the boiler drum, and the water level in the boiler drums to automatically control the fuel oil valves furnishing fuel oil to the burners, feedwater valves furnishing water to the drums and forced draft burners furnishing air to the burners. All of the pneumatic components of the ABC are well defined mechanical components which have predetermined performance criteria in order to provide the control characteristics required by specifications. Because the components of the pneumatic control system are interrelated and perform in a cascading mode, misalignment malfunction or degradation in the operating characteristics of the components of the system can cause total degradation without any clear indication as to which component is causing the problem.

The current diagnostic tools available to assist ship's force in the maintenance of the system are contained in the On-Line Verification (OLV) Procedures. This manual walks the operator through a series of steps in the performance of High Power, Low Power steady-state tests and step change response tests of selected controllers. During the test the operator compares the obtained readings with predetermined acceptable values. In addition, the Boiler Flex test is performed at

predetermined intervals to evaluate the ABC's performance during a 70% up or down ramp.

The Fireroom Maintenance Management Support System (FMMSS) was designed to allow execution of the proven ABC diagnostics in a more expeditious manner and provide additional capabilities in support of system maintenance. In addition, with the simultaneous monitoring of all controller input and output values during transients, a true dynamic performance evaluation can be made of both components, as well as the total system. With the availability of all control signal values, simple display formats of the cascading input/output signal levels allow a qualified technician to easily scan for improper values.

With application of on-line data acquisition and diagnostic analysis, the testing and the monitoring of the pneumatic control system is performed in a much more expeditious manner. In line transducers are provided which measure the pneumatic pressures in the pneumatic control system as well as some of the parameters in the boiler system itself. These transducer outputs are converted to digital values and are read by a computer every tenth of a second. The computer is controlled by a touch sensitive display and is programmed to provide directions on the screen to lead an operator though each of the series of test steps necessary to evaluate the various components of the pneumatic control system. In addition, the system continuously monitors the various parameters of the system and, through control of the touch sensitive display, the operator can have displayed to him the various parameters of the system grouped in logical arrangement (fuel loop, air loop, feed loop). The system is able to continuously monitor the sensed parameters of the equipment to determine whether any of them are outside of specification values and to indicate to the operator which parameters require adjustment. During the FLEX test, all of the parameters are read at one tenth of a second intervals, stored and is compared in a statistical analysis against an on-line mathematical model to obtain a figure of merit for the operation of the major components of the pneumatic control system.

Through the use of FMMSS, the pneumatic control system can be tested much more expeditiously and thus, maintenance of the system is greatly facilitated.

3.5.a <u>System Configuration</u>

FMMSS consists of three basic components:

- o A microcomputer
- o A plasma display
- o A set of sensors

The microcomputer initially installed on the USS T.C. HART (FF 1092) was a 5 MHz Intel 8088 8/16 bit data path microprocessor based system with 256K RAM (Random Access Memory) and 256K ROM (Read Only Memory). The STD bus based microcomputer also contains cards that allow for data acquisition, data transmission and reception, a clock, and a magnetic bubble drive. The bubble drive is used to store system configuration data and data from some of the tests that FMMSS runs. This data can be used to determine expected component failure by trend analysis.

An 8087 coprocessor off loads the main processor for mathematically intensive computations. Communication to external devices is provided by 4 RS232C channels. A General Digital VPII plasma display is connected to the computer via one of the RS232C channels. The display consists of a 40 character wide by 12 line high alpha-numeric touch sensitive plasma display. The infrared interactive plasma display was chosen, since data entry requirements were minimal and installation of keyboard was not readily feasible. The plasma allows the user to input the required data and change the display pages by interrupting the path between infrared transmitters and receivers mounted in the screen frame.

To support practical application without intrusion into the existing system and its operation, the FMMSS was designed to interface to pneumatic and electric signals that are available at existing test points in the control console. Table 1 lists the interface points which allow system monitoring and performance analysis. The pneumatic signals are converted to a voltage equivalence using pressure transducers with sufficient qualifications. The converted signals are conditioned and interfaced to the computer which samples the values at up to 10 times per second depending on the program being executed. The acquisition of the signals provides for passive monitoring of control signal values without affecting ABC performance. The availability of the monitored parameters allows the assessment of control component alignment/conditions with respect to specifications.

The system consists of operating system code and application code. The operating code (ERTOS) was custom designed for this type of application. All functional features of a real-time, multi-tasking operating system are available to support FORTRAN or C based application programs.

The application code is structured in a modular format to allow ease of testing and modification in support of enhancement. The existing modules include:

- Monitoring module (MM)
- OLV fault tree module (OLV)
- Dynamic response analysis (DRA)
- Data recording (DR)

3.5.b <u>System Capability</u>

The functional capability of the FMMSS can be grouped into four categories:

- System monitoring
- OLV test performance
- Continuous OLV monitoring
- Dynamic performance/evaluation

The sensors change a pressure signal into an electrical signal and allow the system to perform its readings almost instantly. FMMSS uses the signals to perform the standard Navy OLV and FLEX tests. These tests aid the operator in detecting components that are not performing correctly. The OLV and FLEX tests were translated from the FF 1052 technical manuals onto the FMMSS computer. Since FMMSS knows how to perform each test, and can read all vital signals through the sensors, the system can perform these tests more efficiently than ships maintenance personnel. This can save valuable time and resources and allow each of the tests to be run more frequently. In addition, since FMMSS reads all the signals virtually simultaneously, the plant status information represents more accurately the interrelation of the data than can be accomplished with operator logging of all the parameters.

The controllers used aboard the ship are pneumatic proportional plus integral controllers. The proportional control yields an output that is directly related to the input. For example, if the input is 1 psig and the output is 4 psig, an increase of 1 psig to the input would cause the output to change to eight PSI. In this case, the output was four times the input. Four was the proportional gain. Integral control slowly eliminates error from the output. This can be accomplished by detecting the difference between the desired output signal and the actual output signal. In the previous example, the output was supposed to be eight PSI. But, if the proportional gain allowed the output to reach only seven and one-half PSI, there would be an error in the output signal. To eliminate this half PSI error, integral control would continuously subtract the desired signal from the actual signal and add that to the output. This would cause the output error to become smaller, so the next addition would be just slightly smaller. Slowly, the output would be exactly eight PSI.

FMMSS uses a computer model of the proportional plus integral controller to evaluate the performance of each component in the boiler control system. This allows FMMSS to calculate the actual gain (proportional) and reset (integral) values of the controllers.

FMMSS acquired data from the sensors stored in memory for future use. Following a six month deployment of the USS T.C. HART (FF 1092) stored data was used by shore-side maintenance organizations to determine ABC system performance and determined maintenance requirements. Data from the whole class of ships can be used to track design problems and provide insight for corrective action.

3.5.c Function Description

3.5.c.1 System Monitoring

This function allows display of raw control system pneumatic pressure data in a logical grouping. The parameters listed in Table 1 represent the available real-time monitoring data base from which the areas of interest were grouped. The fuel/air control loop, feedwater control loop, or the master control loop can be requested for real-time snapshot display. A trained ABC technician could use this for a quick check whenever instabilities are evident.

3.5.c.2 OLV Test

This feature allows call-up and sequential execution of any one of the ten OLV tests presently available in the OLV Manual. Since some of the system alignment information and set up requirements require operator action, the tests are accomplished interactively. The computer has access to the ABC control signal values, hence all of the comparisons and performance evaluations are done on-line and the results are displayed to the operator along with appropriate maintenance recommendation. The OLV logic trees, as presently implemented in the fleet, were programmed into the processor which allows a more time efficient execution of the tests.

3.5.c.3 COLV

Continuous on-line verification continuously monitors the entire system to look for faults while the system is in normal monitoring mode. This test is performed without operator interface and consists of the same OLV tests. When a fault occurs, a warning message is displayed at the bottom of the monitoring screen.

3.5.c.5 Dynamic Response Test

When the FLEX test function is called up, the FMMSS precalculates end point power level using the start point value selected by the operator. During the up ramp or down ramp, the system records the critical boiler parameters, boiler pressure and water level, 10 times per second as well as the input and output signals associated with each control system component instrumented. Once the test is completed, the performance of the total system with respect to transient and steady state specifications is assessed and displayed to the operator. If a diagnostic is requested by the operator, the dynamic performance of the control components is compared with the performance of mathematical models of correctly adjusted components using the same input values generated by the actual system. A statistical correlation of the real component and its model during the same transient results in a "figure of merit" ranging from 0 to 1. Satisfactory, Marginal and Unsatisfactory figures of merit have been computed for all controllers based on test results as well as practical experience of engineers and technicians involved with these ABC systems. The time based dynamic transient data is recorded and can be off-loaded to an external device like a printer or portable test equipment for off site analysis. For unsatisfactory component performance, an applicable OLV test can then be run to identify adjustment requirements.

The FMMSS is installed in USS T.C. HART (FF 1092). Upgraded 80C186 versions are also being installed in USS AMERICA (CV 66) and USS WASP (LHD-1) along with the rest of the PPCAS for complete monitoring and expert diagnostics of the entire steam plant system including boilers, main engines, forced draft blowers, main feed pumps, and condensers.

3.6 Diesel Engine Monitoring and Analysis (DEMA)

The PPCAS application for diesel engines is called Diesel Engine Monitoring and Analysis (DEMA). The DEMA System similar to the systems installed onboard inland marine vessels was also installed onboard the USS HARLAN COUNTY in October, 1987 and was recently removed after evaluation. This system consisted of sensors installed on 1A, 1B, and 1C Main Propulsion Diesel Engines (MPDES) on the starboard shaft, a computer and display screen installed in NR1 Engine Operating Station (EOS) and a log printer installed in the adjacent interior communication workshop. A list of data acquisition points for the applicable equipment is listed in Table 2. The performance degradation trigger points and scan groups of applicable data points to be recorded at the inception of performance degradation were developed by the ship's engineering department personnel in conjunction with Navy headquarters diesel engineering personnel.

The performance alarm scan groups are listed in Table 3.

The DEMA Computer Program and display allow the Engineering Officer of the Watch (EOOW) and Engineman of the Watch (ENOW) to simultaneously observe the response of all indicated MPDE parameters. These watchstanders can instantly identify faulty components, out of specification parameters or impending casualties. With DEMA, he can observe the entire engine system responses at one time on the DEMA terminal inside the EOS. The ENOW can identify and act on impending and/or actual casualties much quicker and with much more confidence, thereby reducing the period that the affected shaft may be operating at reduced capability.

During various operational deployments, the ship's force found that the DEMA Trend Program performed much faster and more accurately than the manual method. The time difference is 30 seconds vice two hours. This instantaneous collection of engine data made all collected information more accurate and was not affected by sporadic load and sea state changes. Ship's operating personnel recommended that in the future, the DEMA equipment and program be modified so that the system will record all the required reading (i.e. cylinder firing and compression pressures) thereby eliminating the need for ship's force personnel to manually conduct cylinder trend analysis readings.

While not substantiated by quantitative results, it was the opinion of the ship's engineer that the watchstanders operated the plant more efficiently with the display console available. The operators quickly gained familiarity with the DEMA system and grew confident with its operation. The system was taught to the throttleman (used in load balance for full power run) and Oiler/Msgr (used to verify manual readings). This resulted in all watchstanders learning and understanding the interrelationship of the various main and auxiliary systems. Initially, the operators used the exhaust temperature display for main propulsion diesel engine load balance and sharing. This is a critical element when using multi-engines per shaft with regards to fuel savings (load balance), reduced engine maintenance and head changeouts (burned valves caused by high and out of specification exhaust temperatures). All questionable pressure and temperature parameters taken by the Oiler/Msgr were verified with the DEMA and investigated by the watchteam. The following casualties are examples which were flagged by DEMA and noted by the operators within seconds of their occurrence:

a. LOSS OF SALT WATER PRESSURE ON 1C MPDE - Emergency cooling was cut in to prevent system and equipment overheating and engine was secured. Investigation revealed a sheared pump shaft.

b. LOSS OF SCAVENGING AIR 1A MPDE - Simultaneous decrease of scavenging air and increasing in cylinder exhaust temperatures were noted on the display screen. These changes were noted and verified on the watchstander's log and the engine was secured.

c. EXHAUST TEMPS ON 1A, 1B, 1C MPDES - The display would frequently indicate out of specification exhaust temperatures (too high or too low). Each time the affected engine readings were verified by the hand held equipment and proven correct. Each corrected reading is a savings in material expended and equipment wear and tear.

Automated trend data recorded on the magnetic bubble cartridges were mailed back for analysis monthly. The initial data reductions indicated that the steady state criteria for trend data logging was too loose which allowed occasional data logging during transient conditions. Following required adjustments, trend data became more meaningful. In addition to trend data, performance alarms, and logged scan groups with time/date stamps were also contained on the bubble cartridges mailed back.

The prototype testing results to date have provided positive results relative to the application of a shipboard maintenance computer with user oriented data acquisition and recording capabilities. With the added feature of embedded expert diagnostics, it provided both operator training as well as timely maintenance support in advance of reaching a lower level of equipment degradations.

3.7 DEMA to ADETA Interface

As a result of the positive USS HARLAN COUNTY (LST 1196) evaluations, a preliminary technical baseline for the DEMA system was established by the Navy incorporating additional features such as fuel measurement, turbocharger parameters, and cylinder firing pressure.

The MACHALT baseline DEMA system will be installed in USS GERMANTOWN (LSD 42) at the end of 1990 as an integrated DEMA package for in-depth Navy pilot program evaluation.

Major maintenance benefits can be derived by integrating and interfacing the DEMA System with the Navy's Automatic Diesel Engine Trend Analysis (ADETA) program described in section 4. A Navy task effort was performed to evaluate and demonstrate the interface options. Since DEMA is an on-line continuous monitoring system that collects and stores condition data, automatic collection of the required trending parameters for the

ADETA System will greatly enhance the data collection process and provide accurate, reliable data to the fleet operators and technical maintenance engineers.

The DEMA to ADETA interface has been designed including interface data formats, file structure/protocol, and type of data. The DEMA software has been modified to create an output file providing ADETA trending parameters and is compatible with the Navy's Zenith PC/AT computer that ADETA resides in. The DEMA/ADETA system integration/interfaces will be technically verified on an actual operating ship, the USS GERMANTOWN (LSD 42), under the Assessment of Equipment Condition Program pilot efforts sponsored by the NAVSEA Surface Ship Maintenance Division, NAVSEA Diesel Systems Engineering Division, and the Naval Ship System Engineering Station (NAVSSES).

3.8 Projected Benefits of an Integrated PPCAS/ADETA System

There are many potential cost benefits of PPCAS implementation which are greatly enhanced by interfacing the PPCAS and ADETA systems for monitoring and diagnosis of Naval systems. These benefits include:

- Increase platform readiness through increased system availability
- Reduce maintenance actions
- Reduce inspection and repair man-hours
- Reduce I, D level work by more effective use of O level resources
- Extend equipment life/overhaul cycle
- Reduce spare part provisioning
- Reduce maintenance induced failures
- Reduce DFS costs per ship/year
- Improve O level equipment management effectiveness
- Reduce recurring costs/year to manage fleet OMN/OPN $

4.0 AUTOMATED DIESEL ENGINE TREND ANALYSIS (ADETA)

Diesel Engine Trend Analysis, commonly called Trend Analysis, requires recording and plotting selected engine

parameter data against engine operating time. This results in a graphical representation of engine operation that can be interpreted to determine the condition of engine internal components. Trend Analysis, which has been in use in Navy applications since the late 1960's, represents a very logical method of determining the need for major engine maintenance actions. It is a middle of the road approach between the extreme cases of :

- o Running an engine until something breaks and then replacing the broken part(s), or

- o Disassembling the engine for parts replacement so often that components never even approach the manufacturer's maximum wear limits.

Trend Analysis is more efficient than either calendar time or operating time directed maintenance. Calendar time requirements usually result in under or over maintenance; operating time directed maintenance does not compensate for different operating conditions which can also result in under or over maintenance.

4.1 *DEVELOPMENT OF TREND PHILOSOPHY*

The development of the diesel engine trend analysis philosophy was originally driven by a need identified by the submarine forces. The nature of the installations caused submarine diesel engine overhauls to be expensive both in dollars and time. In addition, there was no valid method of determining engine condition between overhauls other than "open and inspect". Because of the casualties that followed, an "open and inspect" evolution and the desire to reduce overhaul cost by stretching the time between overhauls, the submarine force requested development of a program to eliminate these problems.

Selecting the parameters to be recorded for evaluation was at first considered to be a difficult task as a diesel engine is a very complex piece of machinery to evaluate in depth, using operating parameter data. Depending on the engine and support systems, well over 100 separate parameters can be recorded during operation. Since the intent was to determine the engine condition, not the system condition, only those parameters that were readily obtainable and would help define the power producing capability of the engine were selected for evaluation. The following parameters were selected:

- o Cylinder firing pressure (for each cylinder)
- o Cylinder compression pressure (for each cylinder)
- o Cylinder exhaust temperature (for each cylinder)

- Fuel injection pump rack or governor power piston position
- Crankcase vacuum
- Lube oil pressure at engine inlet
- Scavenging air pressure
- Lube oil consumed per 100 hours of operation

After the parameters to be analyzed were selected, the parameters were plotted (each on an individual graph) versus each 100 engine operating hours as follows:

- Average, high and low values for cylinder firing pressure
- Average, high and low values for cylinder compression pressure
- High and low values for cylinder exhaust temperature
- Governor power piston position or average injection pump rack reading
- Crankcase vacuum
- Lube oil pressure
- Scavenging pressure
- Lube oil consumption

Consideration of typical ship operating profiles lead to the selection of 80% load as being the most practical trend data collection point.

4.2 USE OF TREND ANALYSIS PROCESS

The Trend Analysis process consist of utilizing engine operating data to make graphs that show at a glance the condition of the engine. This graphical record allows a long term evaluation that specifically gives ship's personnel:

- Visual information that quickly shows any abnormal readings to indicate a developing problem.

- Combinations of out-of-specification readings to help pinpoint impending failure of parts.

- Detection of gradual degradation of overall engine condition for determining overhaul times.

After all required reading have been obtained, recorded and plotted, they must be analyzed to determine and need for maintenance or corrective action. The graphical data is reviewed and analyzed both:

- for obvious large changes in values from the last data point and

- for the overall shape of the curve or trend; thus the name of the process - Trend Analysis.

Collection and analysis of diesel engine trend data does not require the use of special equipment, complex mathematical formulas or equations. However the analysis is experience sensitive; a knowledge of engine operating parameter data and the relationship among the parameters is required. The person analyzing the data must be aware of and knowledgeable with the "cause and effect" relationships when reviewing the graphs. A proper analysis is critical as all maintenance and overhaul determinations are made at this stage of Trend Analysis.

4.3 AUTOMATING DIESEL ENGINE TREND ANALYSIS

The NAVSEA Diesel Engine technical code has developed an automated program that utilizes an expert system to provide ships force with the "cause and effect" relationships necessary to properly analyze the engine.

Before the Fleet wide Automated Diesel Engine Trend Analysis Program could be implemented, the following was accomplished:

- A complete review and revision of all of the diesel engine Planned Maintenance Subsystem (PMS) in accordance with the Navy Maintenance and Material Management (3M) Systems.

- Development of a process and procedure to collect, analyze, manage and store all of the data required to accomplish trending of all Fleet diesel engines subjected to Trend Analysis.

- Preparation of a Chief of Naval Operations Instruction to define and tie the system together. This instructions defines all involved activities' responsibilities, data sheets, data taking frequency and procedures along with a complete definition and explanation of the Automated Trend Analysis Program.

The following paragraphs outline the methodology that has been used to develop the most complex part of the Trend Analysis improvements - the process and procedures to store, manage and analyze the data required to automate the Trend Analysis process. The data base that is being built by the ships that were issued the Automated Trend Analysis program is not only valuable to Fleet operators and maintenance personnel, but to the diesel engine technical community. The Diesel Engine technical code is

currently developing programs to trend diesel engine inspections, including clearance data. The inspection data trends will be used to indicate possible problem areas for more in-depth analysis.

Remote personal computers are being utilized to enter, edit, accomplish data analysis of trend data, and then format and transmit the data to the NAVSEA Diesel Engine Management Information System (DEMIS) network. DEMIS creates, maintains and enhances the data bases that manage the trend data performs, engine/fleet population trend analysis and evaluates results of the trends.

The data entry process at the remote shipboard PC has been designed to be as completely menu driven as possible. The data entry, editing, local analysis and transmission system is operable by persons with little or no engine or computer experience. After the system start-up and sign on process is completed, the operator is prompted for a ship's hull number. A listing of all diesel engines on that ship, along with a data mode menu will be presented. The operator is then prompted to select an engine and a choice of data modes from - enter data, transmit data or review trend data. The effort that has been put into the system design to ensure ease of operation is best shown by the following design guidelines for the data entry mode:

- When the enter data mode is selected, either a warning that previously entered data has not been transmitted will be issued with a return to the main menu, or the operator will be prompted to enter specific operating data for the engine selected. To minimize input key strokes and reduce the need for a decision on the part of the operator, only data unique and in the terminology peculiar to the engine will be requested. For example if the engine is a 12 cylinder unit, only 12 cylinder exhaust temperatures will be requested. After each group of data is entered, it will be checked for completeness and accuracy, i.e., missing or out of expected range values. If any data appears to be invalid, an error message requests the operator to check/correct the data. If the corrected data still appears to be invalid, and error message results with a menu of possible reasons for the data errors. The operator must select or otherwise provide explanations for the system to accept the data. When a complete data set has been entered, the data is compared with the three previous submissions (that are always resident in the remote PCs). If the overall trend varies by more than about 15-20 percent from the previous data, an error message will indicate to the

operator that data may not have been taken under proper conditions. Data is stored in a temporary file until its correctness can be verified. If a group of parameters indicate a possible impending problem, a warning to that extend will be printed on the screen and to a special warning files. When all input parameters have been satisfied, the operator is given the choice of formatting and saving the data for later transmission or formatting with immediate transmission to DEMIS. If immediate transmission is selected and the previous data set has not yet been transmitted, an error message is generated, the data formatted and saved with a return to the main menu. Mo more than two sets of data is accepted by the remote PC until transmission to DEMIS is up-to-date.

The design of the resident program within the PC to manage, compile and analyze the trend data has also received detailed attention. In addition to accomplishing the Trend Analysis, these less obvious requirements have been addressed in the system design:

- Provisions to maintain data on an engine from overhaul to overhaul. When an engine is overhauled, the complete set of trend data must be taken from the active file and placed in a historic file and a new trend started. A provision in the program was also included to discard existing data and start a new trend if a ship's engine is replaced due to an extensive casualty or ship improvement program.

- An extensive data protection scheme was developed. A local command can only have access to the trends for engines under their control. Once data is entered into DEMIS, it can only be reviewed from the remote PC in trend form; it cannot be changed from the remote PC. Provisions were made for transfer of a ship from one command to another. The engines' trends are maintained, but, the reporting and reviewing command will change.

- Procedures were developed to assist in enforcing the Trend Analysis Program requirements. Provisions were made in the analysis process to flag repeated erroneous, missing or out of specification parameters after no more than three occurrences.

- Provisions for future expansion without reformatting the entire data base are included. The capability to add data for each engine that will more than double the

data base size were accommodated.

4.4 ADETA SUMMARY

A detailed examination of the diesel engine trend analysis was conducted. When the basic concept proved to be still sound, the decision was made to update the venerable process by utilizing state-of-the-art analysis techniques. The result is starting to exhibit a decrease in both diesel engine maintenance actions and repair costs with an over all increase in diesel engine operational readiness. Operating personnel are also having an increased confidence in their engines' capabilities.

The addition of diesel inspection data will give engineers data form a program designed for and by engineers. It will put them in a proactive rather than a reactive position by making data available with a key stroke as part of the DEMIS network.

5.0 CONCLUSIONS

The surface navy to date has made a significant progress in adapting the concept of Condition Based Maintenance. For the concept to be effective, it is imperative that the ship operators be able to determine machinery condition reliably and with confidence, at an affordable cost. It is our opinion that with the use of "Non-Intrusive" on-line monitoring and diagnostic systems, the surface navy could maintain high operational readiness with fewer people and fewer dollars not only of the current active fleet but use of such onboard systems will be essential for ships of the 21st Century.

The Department of Defense (DoD) Ship Operational characteristic Study (SOCS) has identified Condition Based Maintenance technology as an essential element of ship operational characteristic and design for surface combatants of year 2010 and beyond. NAVSEA's efforts described in this paper is a step in that direction, to demonstrate the feasibility of tailoring commercially available off-the shelf technology for use of surface ships.

Even under severe funding constraints, using Non Developmental Item (NDI) approach for system acquisition, the Navy could revolutionize current concept of ship maintenance. The implementation of condition based maintenance will require a new philosophy of system and equipment design, repair procedures, and maintenance training.

Ref. NR	Description	BLR 1A	BLR 1B	Common
1.	Drum Pressure	X	X	
2.	Steam Pressure Transmitter Output	X	X	
3.	Air Flow Transmitter Output	X	X	
4.	Steam Flow Transmitter Output	X	X	
5.	Feedwater Flow Transmitter Output	X	X	
6.	Drum Level Transmitter Output	X	X	
7.	High Signal Selector Output			X
8.	Steam Pressure Controller V.C.			X
9.	Steam Pressure Controller Output			X
10.	Boiler Master A/M Station Output	X	X	
11.	Fuel Air Ratio Station Output	X	X	
12.	Air Flow Controller V.C.	X	X	
13.	Air Flow Controller Output	X	X	
14.	Steam Flow Rate Relay V.C.	X	X	
15.	Steam Flow Rate Relay Output	X	X	
16.	Range Modifier Output	X	X	
17.	Low Signal Selector Output	X	X	
18.	Characterizing Relay Output	X	X	
19.	Combining Relay Output	X	X	
20.	Drum Level Controller V.C.	X	X	
21.	Drum Level Controller Output	X	X	
22.	Feedwater A/M Station Output	X	X	
23.	F.D. Blower #1 RPM	X	X	
24.	F.D. Blower #2 RPM	X	X	
25.	Fuel Oil System Pressure	X	X	
26.	Fuel Oil Burner Pressure	X	X	
27.	Feedwater Header Pressure			X

<u>ABC System Interface</u>
<u>Table 1</u>

Each Main Engine

Engine RPMs
Rack Position
Cylinder Temperatures 1 thru 16
Stack Temperature
Salt Water Injection Temperature
Salt Water Overboard Temperature
Jacket Water Temperature to Engine
Jacket Water Temperature from Engine
Salt Water Pump Pressure
Jacket Water Pump Pressure
Lube Oil Pump Pressure
Lube Oil Header Pressure
Lube Oil Filter Outlet Pressure
Lube Oil Strainer Inlet Pressure
Turbo Charger Lube Oil Pressure
Fuel Oil Pump Pressure
Crank Case Vacuum
Air Manifold Pressure
Air Intake Depression
Air Intake Manifold Temperature
Air Intake Manifold Temperature
Turbocharger Air Discharge Temperature

Other Plant Parameters

Propeller Pitch
Main Reduction Gear Lube Oil Pressure
Engine Room No. 1 Heat Stress Temperature

USS HARLAN COUNTY (LST 1196)
SENSOR SUITE
Table 2

Entire Engine Group
Low Fuel Oil Pump Pressure
Low Fuel Oil Header Pressure
Low Fuel Oil Pump Pressure
High Lube Oil Filter Differential Pressure
Low Turbocharger Lube Oil Pressure
Low Main Reduction Gear (MRG) Lube Oil Pressure
Low Lube Oil Header Pressure
High Lube Oil Temperature to Engine
High Lube Oil Temperature from Engine
Low Salt Water Pump Pressure
Low Jacket Water Pump Pressure
High Jacket Water Temperature to Engine
High Jacket Water Temperature from Engine
High Cylinder Exhaust Temperature
Cylinder Exhaust Temperature Differential
High Crankcase Vacuum

<u>USS HARLAN COUNTY</u>
<u>PERFORMANCE ALARM SCAN GROUP</u>
<u>TABLE 3</u>

PROPULSION PLANT CONDITION ASSESSMENT SYSTEM (PPCAS)

SHIPBOARD SYSTEM CONFIGURATION

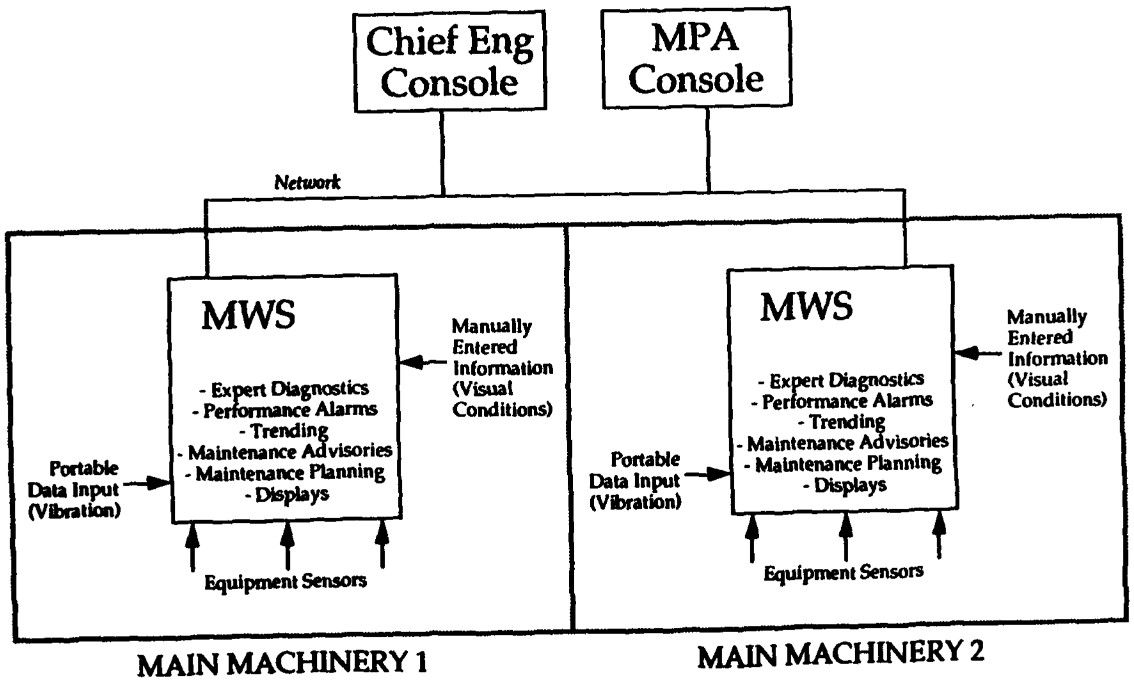
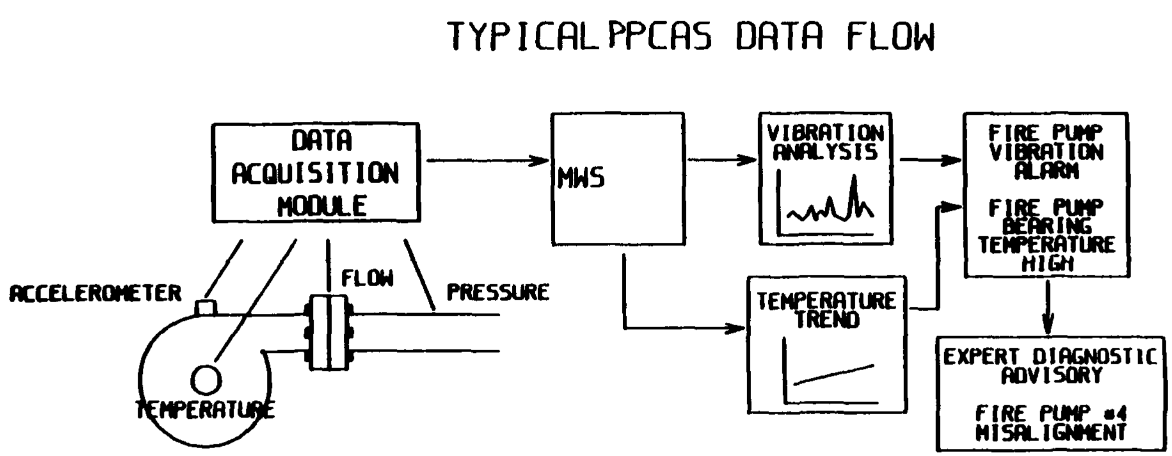


FIGURE 1



TYPICAL PPCAS DATA FLOW
DATA ACQUISITION MODULE
ACCELEROMETER
FLOW
PRESSURE
TEMPERATURE
MWS
VIBRATION ANALYSIS
TEMPERATURE TREND
FIRE PUMP VIBRATION ALARM
FIRE PUMP BEARING TEMPERATURE HIGH
EXPERT DIAGNOSTIC ADVISORY
FIRE PUMP #4 MISALIGNMENT

FIGURE 2

AUTOMATIC CONTROL SYSTEM FOR PROTOTYPE SHIPBOARD NITROGEN GENERATOR

by Abdi Nazari (NAVSEA 56Y14),
Jack L. McCrea and David G. Barr
(Westinghouse MTD)

1. ABSTRACT

This paper describes the automatic control system currently being used on the prototype shipboard nitrogen generator and discusses proposed systems to be used for future production units.

Cryogenic separation methods currently being used to produce oxygen and nitrogen are complex, require intensive manpower, and exhibit low reliability. To address these problems, NAVSEA initiated the design of a prototype nitrogen generator to produce gaseous nitrogen at up to 99.5 percent purity using a pressure swing adsorption process. The prototype generator can produce 100 lb/hr of nitrogen at 5000 psig. The generator consists of a low-pressure air compressor, a pressure swing adsorption unit, and a high-pressure compressor, that together take air and separate the nitrogen using pressure swing adsorption. NAVSEA plans to use the generator on aircraft carriers and subtenders to replace cryogenic separation methods.

The prototype nitrogen generator has an automatic control system which makes use of a programmable logic controller (PLC). Background is given on the selection of a PLC for the prototype control system and details are provided on the controller selected. Future control methods such as a SEM-based (standard electronic module) controller, dedicated controller or military qualified PLC are also discussed for the production model nitrogen generators now being planned.

2. INTRODUCTION

Shipboard requirements for nitrogen are currently being provided by cryogenic separation methods. In this process, air is liquified and separated into liquid nitrogen or oxygen in an O_2N_2 producer plant. The liquid is then stored in liquid storage tanks. All liquid nitrogen produced is vaporized and stored in 5000 psig gaseous storage flasks. When and how much liquid nitrogen is produced and vaporized is determined by the ship's force depending on the system demand for nitrogen. Operation of the separation,

storage, and vaporization equipment is manual. This equipment is generally complex, and requires advanced operator training and intensive manning. The equipment, especially the oxygen/nitrogen producer plants, has demonstrated low reliability.

In 1983, the Gas Processing and Cryogenics Branch of NAVSEA (56Y14) initiated an in-depth study on alternative methods for generating nitrogen for shipboard use. Several alternatives were investigated including: improvement and automation of the existing cryogenic producers, hollow fiber membrane separation, and pressure swing adsorption. It was determined that existing cryogenic separation technology could not be improved in efficiency or cost-effectiveness and membrane technology at the time proved to be ill-suited. Molecular sieve pressure swing adsorption (PSA) was recommended as the most suitable and promising technology for nitrogen generation.

Based on the results of this study, laboratory tests were performed on a small scale PSA system to establish process flow and design parameters (1). Next, a pre-prototype feasibility model was fabricated and tested utilizing PSA technology. This unit produced 25 lb/hr of nitrogen at 99.5 percent purity, meeting all requirements. The model was installed on the USS RANGER in 1987 and is still operating.

Shipboard nitrogen requirements were evaluated by NAVSEA and it was determined that a nitrogen generator capable of producing 100 lb/hr of gaseous nitrogen at 5000 psig and 99.0 percent purity and greater than 50 lb/hr at 99.5 percent purity would be required. It was decided that the generator would consist of a low-pressure air compressor (Navy Standard STAR), a PSA unit to separate the nitrogen, and a high-pressure air compressor to compress the nitrogen to 5000 psig. The generator would have to be designed to meet all the current shipboard requirements of shock and vibration, be a Navy-owned design, and be capable of automatic operation based on the system demand for nitrogen.

Westinghouse MTD was tasked to design the PSA unit which was to be complete with all controls for the generator and include both compressors. At the same time, David Taylor Research Center (DTRC) was tasked to procure, fabricate, assemble and test the full scale prototype nitrogen generator based on the MTD design. Testing was to include operational (technical evaluation), shock (MIL-S-901), and vibration (MIL-STD-167).

DTRC obtained a STAR (Screw Technology Air Rotary) low-pressure compressor, modified a commercial high-pressure compressor, fabricated the PSA unit, and assembled them into a generator which is currently being tested.

3. NITROGEN GENERATOR DESCRIPTION

The nitrogen generator comprises three major components (see figure 1):

- o Low-pressure air compressor (LPAC)
- o Pressure swing adsorption unit (PSA)
- o High-pressure nitrogen compressor (HPNC)

The LPAC is a standard Navy, oil-free, positive displacement (STAR) compressor providing compressed air to the PSA unit at 120 psig. It has its own local controls and sensors for monitoring process parameters, such as temperature and pressure, but is started, stopped, and monitored by the central generator control system.

The PSA unit consists of two beds (pressure vessels) of carbon molecular sieve material, solenoid-actuated pneumatic valves for bed switching, a gas analyzer to monitor purity and a PLC to control the process. The separation of nitrogen from compressed air takes place in the beds. The openings into the pores of the molecular sieve material are approximately the size of oxygen and nitrogen molecules. Oxygen molecules are smaller than nitrogen molecules, and so diffuse into the pores faster than nitrogen molecules. This diffusion into the sieve material is called adsorption. As the compressed air flows through the bed, almost all of the oxygen, hydrocarbons, and water vapor are adsorbed in the sieve material, and nitrogen exhausts from the bed (see figure 2). While one bed is adsorbing oxygen, the other bed is desorbed by lowering the pressure to near-atmospheric and exhausting the waste gas. The bed is then pressurized and the flow is redirected while the other bed is desorbed. Each bed adsorbs oxygen and produces nitrogen for approximately 55 seconds before recycling.

The HPNC is currently an oil-free, three-stage positive displacement compressor with a maximum capacity of 120 lb/hr of nitrogen at 5000 psig. It has its own local controls and sensors for monitoring process parameters, but is started, stopped, and monitored by the central generator control system.

4. CURRENT AUTOMATIC CONTROL SYSTEM

The controls for the nitrogen generator (including the PSA unit and both compressors) are centrally located on the PSA unit. The PLC is the heart of the nitrogen generator's automatic control system.

A PLC is a digitally operating, electronic controller that uses a programmable memory to internally store instructions that implement specific functions such as logic, sequencing, timing,

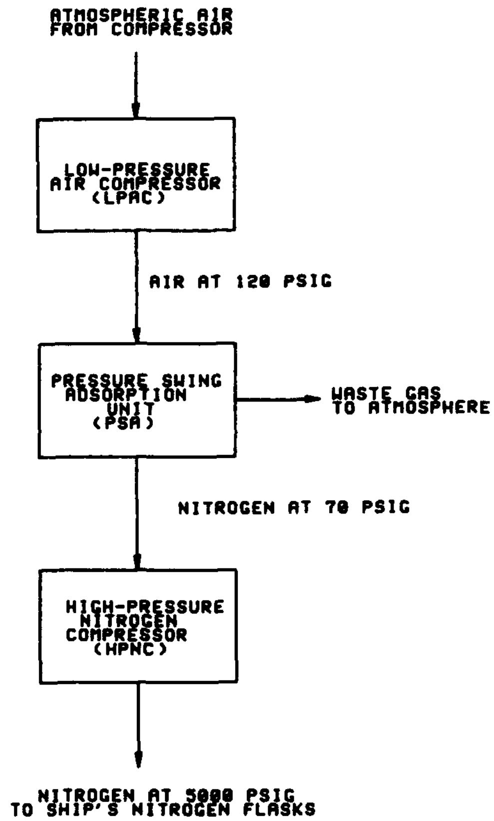


Figure 1. Nitrogen Generator Block Diagram

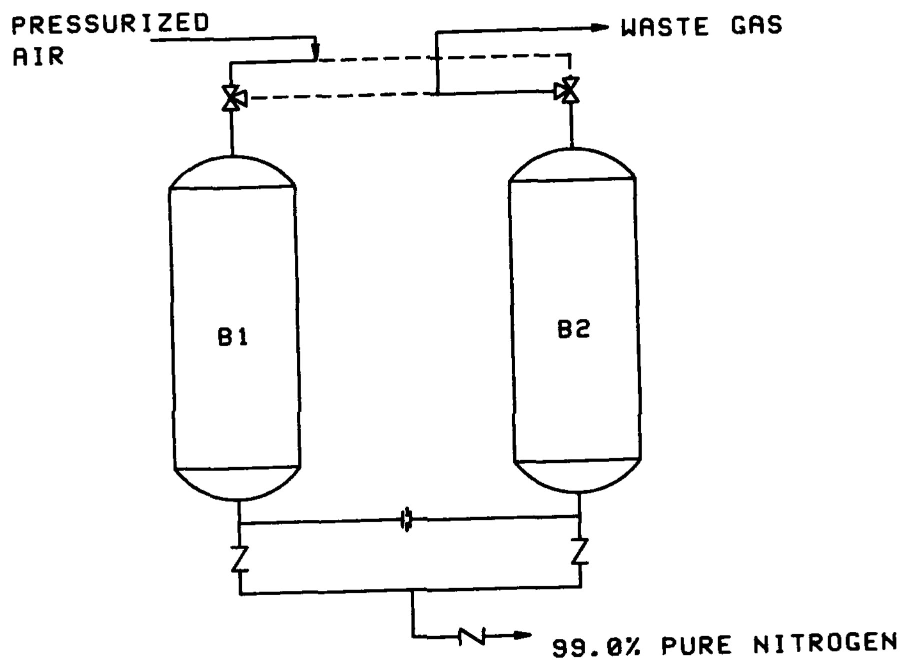


Figure 2. PSA Unit Flow Schematic

counting, and arithmetic to control, through digital or analog input/output modules, various types of machines or processes. A digital computer that performs the functions of a programmable controller is considered to be within this scope. Excluded are drum and similar mechanical-type sequencing controllers (2).

PLCs are typically microprocessor-based and can be considered as dedicated microcomputers. Although other software programs are available, the PLC's programs are generally written in ladder logic, which is very similar to relay logic. Thus, if a relay logic system is replaced with a PLC, the PLC's program can be almost taken directly from the existing relay logic. Ladder logic is generally symbolized as a network of contacts configured to operate coils. This network system is the language of the PLC and is usually represented on a programming terminal's screen display. Generally, the inputs to the PLC are represented by the contacts and the outputs are represented by the coils.

4.1 System Description

The nitrogen generator receives 440 vac ship service power. This power is routed through contactors located in the PSA unit starter panel to the low-pressure air compressor motor and to the high-pressure nitrogen compressor motor. Power is also routed through a transformer in the starter panel where it is stepped down to 115 vac, which powers the PLC, the gas analyzer, and the control circuitry.

The responsibilities of the generator's control system include: valve sequencing, product purity monitoring, ventilation monitoring, alarm detection, and compressor motor control. The PLC receives various inputs from contact closures of pushbuttons, pressure switches, and auxiliary starter contacts. The gas analyzer also provides an analog voltage signal as input to the PLC. The PLC manipulates these inputs and provides the required corresponding outputs in the form of 115 vac contacts. These outputs go to solenoid valves, lights, contactors, motor start circuits, and an alarm horn. Figure 3 illustrates the nitrogen generator control system in block diagram form.

The gas analyzer measures the purity of the nitrogen output from the PSA unit and sends a corresponding analog signal to the PLC. The PLC compares this signal to the acceptable purity setpoint and if the setpoint is not maintained, sounds an alarm and starts a shutdown sequence.

As mentioned above, the PLC receives discrete inputs from start and stop pushbuttons, cooling water and nitrogen product pressure switches, manual/automatic and input air selector switches, and auxiliary contacts on both compressor motor starters

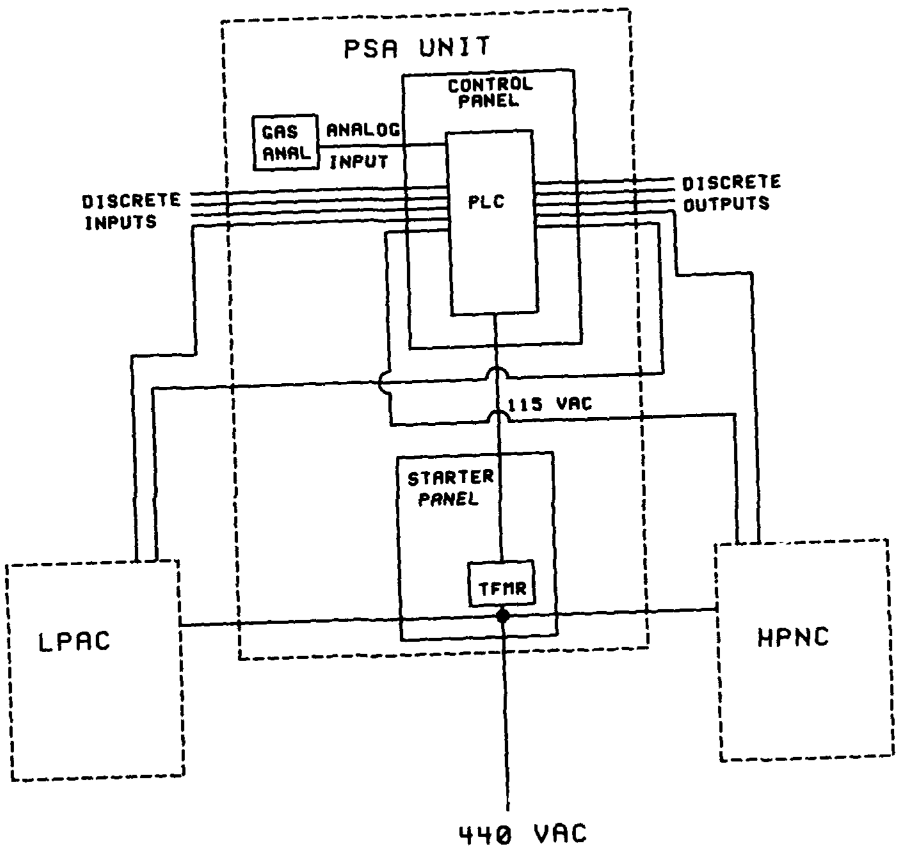


Figure 3. Control System Diagram

and the ventilation system.

The start and stop pushbuttons provide the PLC with the discrete inputs. The PLC will shut down the entire process if the cooling water pressure is lost and will shut down the high-pressure compressor if the output nitrogen pressure is above the setpoint. The manual/automatic selector switch provides the input that allows the system to run in either mode, that is, manual mode in which various portions of the process must be initiated by the operator, or the automatic mode where the operator merely turns the unit on and it runs on its own from there. An emergency mode of operation, using ship's service air system, is also permitted if the low-pressure compressor is not operational. Operator intervention is required (changing a toggle switch position) to initiate this mode of operation, permitting operation without the low-pressure air compressor. The auxiliary contacts on the two compressor motor starters provide feedback to the PLC where the run state of that compressor can be compared to the start or stop requested. The auxiliary contacts from the ventilation system signal the PLC to shut down the entire process on the loss of ventilation.

All of these inputs are used by the PLC to provide the correct outputs to the process. The PLC gives outputs that allow power to flow to the compressor motors and also gives outputs that start and stop these motors. The PLC program uses a series of timers and counters, internal to the PLC software, to sequence the outputs to the solenoid valves. The sequencing of the valves is a repeating 117-second cycle. Outputs from the PLC also light alarm lights and sound an audible alarm in the case of low water pressure, low nitrogen purity, or a contrary signal from one of the compressors.

The discrete input devices are hard wired to the input terminals of the PLC. Since the inputs are discrete, the PLC merely sees a contact change state. A change in state of the input device causes a change of state to all the corresponding contacts in the ladder logic program. The analog input from the gas analyzer is wired to the PLC's analog input module where the nitrogen concentration signal is converted to a digital signal. This digital signal is then loaded into a register in the PLC's microprocessor. The value in this register is then compared to the setpoint value for acceptable output nitrogen purity level.

The outputs of the PLC are directly hard wired to the input devices, such as, lights, solenoid valves, and motor start circuits. An output closure, in the PLC, completes the 115 vac circuit to energize the corresponding output device.

The software program that the nitrogen generator uses is written in ladder logic. The specific software, which was used to generate the actual tailored program for the PLC, was provided by the PLC vendor. The program developed for the nitrogen generator uses input contacts, output coils, timers, counters, internal registers, and internal logic coils and contacts, all of which are part of the PLC's software. There are fifteen discrete inputs, twenty-one outputs, fourteen internal relays, four timers, twenty-five counters, and one analog input used in the program. Figure 4 shows a few typical rungs of ladder logic from the nitrogen generator program.

The program was written as generically as possible by excluding the special software features, such as jump command and master control relay, particular to the PLC vendor software. The ladder logic program was written on a personal computer, downloaded directly to the PLC, and stored in random access memory (RAM). The PLC includes a battery backup for RAM and was also provided with an electronically erasable programmable read-only memory (EEPROM) (which requires no battery backup) where the program was stored for added safety.

A total of 300 logic functions was used in the program requiring 615 bytes of RAM out of the 1024 bytes available to the user and 115 rungs of ladder logic. The program required approximately three weeks to write with a programmer experienced in ladder logic.

4.2 PLC Selection

A PLC was used because it provided flexibility for the prototype nitrogen generator control system. This was especially useful as process problems were worked out of the prototype. The PLC allowed the changing of the timing sequence to fine tune the process without lengthy delays or hardware changes. For example, if it was realized that an existing input, such as a pushbutton, should affect an additional existing output, no additional wiring would be required; it would just be a change to the program. Extra inputs and outputs were provided to allow for any new requirements discovered as a result of testing.

Another advantage of the PLC is the monitoring mode which allows the user to follow the logic flow (on-line) during debugging and verification. This feature of the PLC helped to save time during startup and debugging of the generator as well as during testing and fine tuning of the process aspects of the generator.

The obvious advantages of the PLC are that it is small and lightweight as compared to hard-wiring relays and timers to do the same job. It is also relatively inexpensive compared to relays, dedicated circuit boards, and standard electronic modules.

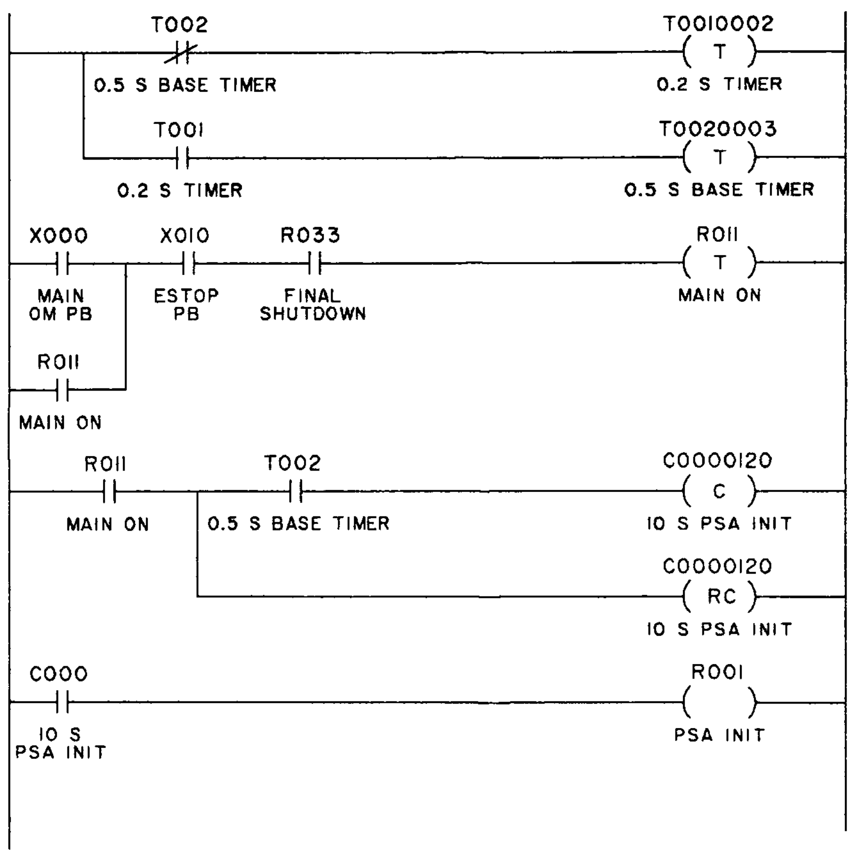


Figure 4. Nitrogen Generator Ladder Logic

4.3 PLC Details

PLCs are common in the commercial world and are produced by a number of different manufacturers. Most manufacturers have models available which use ladder logic and accept both discrete and analog inputs. This nitrogen generator application required a stand-alone controller with 40 to 60 I/O, at least two analog inputs, ladder logic software with timers, counters, EEPROM for program storage, and monitoring capability for prototype debugging and troubleshooting.

Several manufacturers offering the required type of small PLCs were considered. Among those considered were Cutler-Hammer, General Electric, Gould/Modicon, and Westinghouse. The PLC selected for prototype testing provides a total of 80 I/O and can be considered small with an overall footprint of approximately 12 inches by 20 inches by 5 inches deep.

Since all of the small PLCs offer similar features, the specific PLC for the prototype was chosen for the following reasons:

- The PLC vendor was actively working with the Navy to expand MIL-C-2212 to include PLCs.
- The PLC could be modified (hardened) to pass Navy vibration and shock tests and meet schedule deadlines for the prototype generator application.
- The vendor would provide a survey of the PLC's susceptibility to EMI.
- The vendor was currently working to completely militarize the PLC for future requirements of MIL-C-2212.

The final shipboard prototype was fitted with a PLC modified for Navy shock and vibration requirements. This unit was installed on the PSA unit of the generator during shock and vibration testing.

5. FUTURE CONSIDERATIONS

After shipboard evaluation of the generator, NAVSEA plans to procure production models for installation on all aircraft carriers. Before finalizing the design of these production units, several alternative control systems were considered. These were: a militarized PLC, a SEM-based controller and a specially designed controller for the application. Each of these options must meet requirements for shock, vibration and electromagnetic interference (EMI).

The militarized PLC is the most attractive control alternative because of its flexibility (easily programmable), expandability (additional I/O is available) and its low capital cost.

Some of its disadvantages are its limited shipboard repairability and logistics supply support.

The SEM-based controller was attractive because of the standard availability of some cards, it is relatively easily repaired and Navy supply support for most of the SEM cards is already established. The SEM controller, although is not always easily programmed (usually programmed in Assembler), can have a much higher capital cost and usually contains one or more non-standard SEM cards reducing the standard nature of the controller. A controller similar in size was used on the Navy standard HP dehydrator; it was programmed in Assembler, used several non-standard cards and had a higher capital cost than the PLC. Even a SEM controller, although inherently shock-, vibration-, and EMI-resistant, would still require hardware testing before use on the generator.

The special-designed electronic controller is the least attractive of the alternatives. This alternative would include a specially designed and manufactured card or cards mounted and housed to form a controller. It would possibly be more compact than the other alternatives but would not be easily programmed, would not be easily repairable and would have no supply support. Its initial cost, based on preliminary estimates made during dehydrator and generator development, would be less than the SEM controller but still greater than the PLC.

The final decision on the production nitrogen generator controller is to use the fully militarized PLC.

6. SUMMARY

To meet shipboard requirements for gaseous nitrogen, NAVSEA initiated the development, design and production of a prototype pressure swing adsorption nitrogen generator. Central to the automatic controls of the generator is a programmable logic controller. This controller sequences valves, monitors nitrogen purity, controls compressor motors, and detects alarm conditions. The PLC receives inputs from pressure switches, pushbuttons, auxiliary starter contacts, and a gas analyzer and manipulates these inputs and provides outputs to control the generator using ladder logic software.

The prototype nitrogen generator utilized a commerical-type PLC for operational testing. This unit provided the flexibility (through software) to incorporate process and operational changes during debugging and testing of the prototype generator.

A militarized version (one that had been shock and vibration tested) of the commercial PLC was procured and used during shock and vibration of the prototype generator and will be used during shipboard evaluation. This unit is very similar to the commercial PLC and provides the flexibility to make operational changes in response to shipboard requirements.

REFERENCES

(1) M. Leonard, "Shipboard Nitrogen Generating System (SNIGS) Preprototype SVSK 107471-1 Final Engineering Report," Hamilton Standard report no. RN2:87:06, December 1987.

(2) NEMA (National Electrical Manufacturers Association) NEMA Standards Publication/No. ICS 3-1983 Including: Rev. 1-March 1985, PART ICS 3-304, PAGE 1.

COST-EFFECTIVE SPECIFICATION OF COMPLEX MACHINERY CONTROL AND SURVEILLANCE SYSTEMS

by T.McClean B.Sc., C.Eng., M.I.E.E.
YARD Ltd., Consulting Engineers

and M.I.Hawken B.Sc., C. Eng., M.I.E.E.
Sea Systems Controllerate, Ministry of Defence (U.K.)

1. ABSTRACT

Complex, highly automated, software based MCAS systems on modern warships are intended to facilitate operation by reduced manning while cutting Through-Life-Costs. However inadequate MCAS system specifications, reflected by equally poor tenders will exacerbate the evaluation process and can result in time and cost overrun with the procurement of inappropriate solutions. Similar situations occur throughout industry [1].

This paper describes a guide which was prepared to harmonise the specification, tendering and evaluation processes by applying a structured approach to write compatible and correct procurement documentation.

The documentation generated by following the guide will also provide a firm foundation for future MCAS system support and maintenance.

2. INTRODUCTION

There is now general recognition of the importance of specification within defence equipment procurement, but as yet very little guidance on the correct approach to the specification process. This has not raised too many difficulties for the design of Machinery Control and Surveillance (MCAS) Systems, due in part to the :

(a) close collaboration between the specialist section within MOD(PE) and the MCAS System contractor during system development.

(b) use of cost plus contracts and

(c) the size and relative simplicity of previous analogue systems.

However with the advent of more complex software controlled systems, it was recognised that improvements were needed in the specification process.

The need for these improvements was further stimulated by the increasing drive to obtain value for money in defence procurement during the 1980s. This took the form of placing more responsibility with industry and a move toward fixed price contracts. This inevitably changed the relationship between the specifier and controls contractor, requiring a much more formal contract in the form of a complete requirement specification.

It was against this background that MOD(PE) began the development of a specification guide covering the early design stages of MCAS System development.

3. THE PROBLEM

Experience has shown that the majority of errors are introduced at the specification and design stages rather than during system build and test.

This can result in the acceptance of less than ideal systems or corrective action that can cause time and cost overrun, with potential risk to the ship in-service date.

It is therefore preferable to identify problem areas at the outset and take preventative action to avoid them, leading to a more cost-effective and technically acceptable MCAS solution.

The three major areas where errors and problems can be introduced and remain undetected during MCAS system procurement are during:

(a) the preparation of a Specification by the SPECIFIER

(b) the generation of a tender by potential MCAS system VENDORS (one of whom will eventually design, build, supply and fit a system to the ship)

(c) the evaluation of tenders by an EVALUATOR who will compare the tenders received against the specification and determine the preferred solution.

As illustrated in Figure 3.1, specifications are often:

(a) incomplete because many important requirements have been omitted. Sometimes it is assumed that the vendor will fill the gaps - with disastrous consequences

(b) ambiguous - such that the vendor can (and likely will) misinterpret the requirements

(c) inaccurate - simply because they have not been rigorously checked before issue

(d) inconsistent, with incompatible requirements between various aspects of the definition.

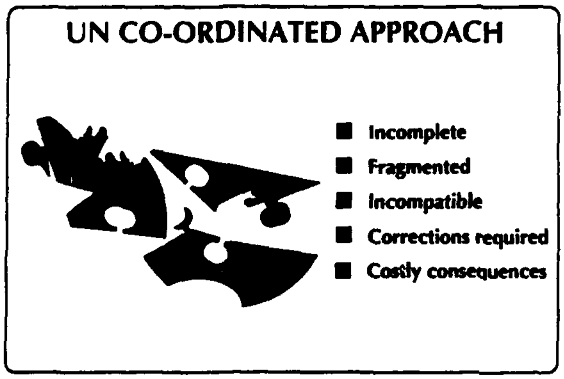


Fig 3.1

Poor specifications generally result in poor tenders in which the vendor:

(a) does not "fill in the gaps" arising from omissions in the specification

(b) offers inappropriate solutions because the specification has been misinterpreted

(c) is not totally aware of the scope of his responsibilities e.g. to complete the requirements definition, to carry out testing, apply quality control etc.

(d) does not provide full supporting information about his company, track record, resources etc.

As a result the evaluator:

(a) will find difficulty in assessing the inadequate tenders against the inadequate specification

(b) may resort to making incorrect and unsupported assumptions about vendors' solutions and capabilities

(c) may, in the extreme recommend an incapable vendor to supply an inappropriate MCAS solution.

Errors identified during these stages will lead to costly re-iteration of documents and unnecessary clarification meetings. Unidentified errors or unconscious acceptance of inadequate documents will certainly lead to problems and perhaps overspend.

4. THE SOLUTION

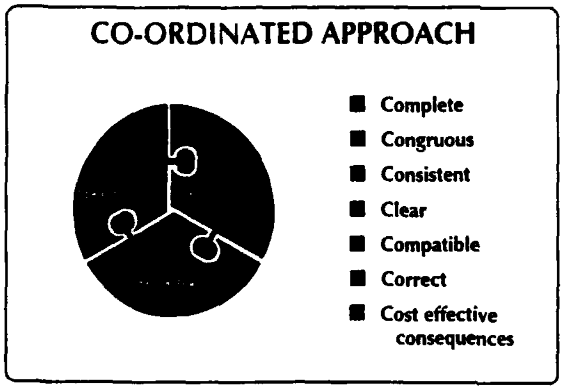


Fig 4.1

The ideal situation is illustrated in Figure 4.1. This represents totally compatible processes in which complete MCAS system specifications are responded to by equally complete tenders offering realistic solutions which the evaluator can easily compare to select and recommend the optimum MCAS system solution.

With these objectives in mind MOD(PE) commissioned a guide which the MCAS system Specifier, Vendor and Evaluator should follow to meet these aims. This guide is known as Sea Systems Controllerate Publication (SSCP) No 27.

This guide needed to take account of the different procurement strategies with which the MCAS System could be procured. These will range from the provision of an MCAS System as part of whole ship competitive tendering to direct replacement of a system during refit. These different procurement strategies place different levels of specification responsibility with the MOD and industry. Therefore the guide needs to cover these differing requirements by providing guidance to both the MOD and industry specifier with different levels of expertise and experience.

5. SSCP 27 CHARACTERISTICS

Before we look at how the Guide is used in a technical sense to steer the specification, tender and evaluation phases towards compatibility it would be useful to note briefly how it was written from a documentation viewpoint. In short, it had to be user-friendly. Specifiers, vendors and evaluators should not be hindered by a Guide that is not immediately understood - they must find it positively helpful.

Considerable effort was applied to achieve this aim by using a logical document structure, concise text, in-text diagrams, extensive use of pull-out charts, tables and pictorial indexes as summarised in Figure 5.1.

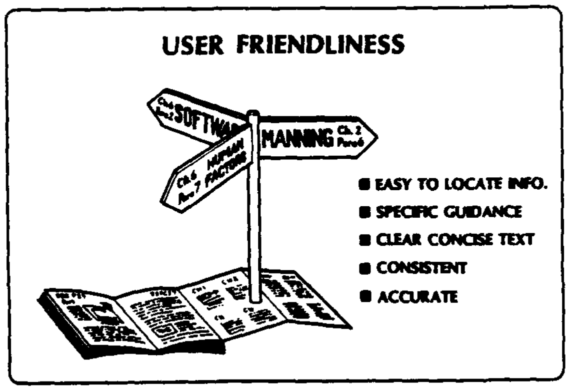


Fig 5.1

6. SSCP27 OVERVIEW

The 5 Part Guide structure is shown in Figure 6.1. Parts 2, 3 and 4 (the Specialist Parts) give particular guidance to the Specifier, the Vendor and the Evaluator respectively, so that they may efficiently progress their tasks to generate the relevant documentation.

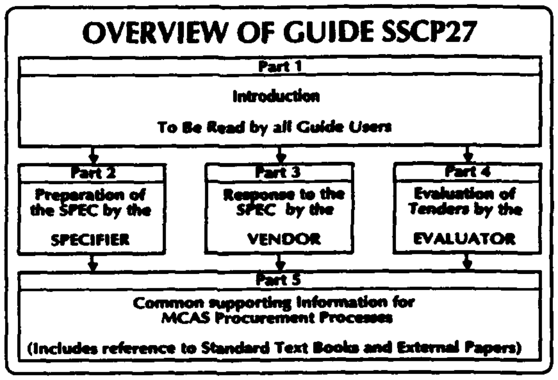


Fig 6.1

Parts 1 and 5 are "common" to all users, providing an Introduction to the SSCP and Supporting Information to those unfamiliar with MCAS.

It is intended that each user only requires his specialist Part together with the brief Part 1 and, if necessary, Part 5.

Before examining each Part in more detail it should be noted that the Specialist Parts 2, 3 and 4 share a common basic structure as summarised in Figure 6.2. Following similar styled introductions each of these Parts outlines the requirements of the document to be generated (i.e. the Specification, Tender or Evaluation Report).

Users who are familiar with MCAS systems may choose to use the diagrams and tables in the Guide as a checklist to ensure that their document conforms with the requirements.

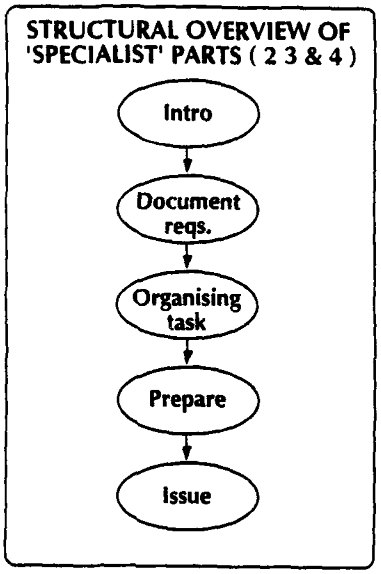


Fig.6.2

The user is then guided on how to organise his team to prepare the document. This is most important as MCAS system procurement involves a multitude of specialist skills. Proper organisation ensures that:

(a) each section of the specification is prepared by an accredited specialist

(b) tender sections are prepared by appropriate specialists

(c) each aspect of the MCAS system tender is assessed against the appropriate part of the specification by the same specialist (in the team) to ensure consistency of subjective judgement.

Step by step guidance is given to write each document down to sub-section level and on how to check and issue the final version.

7. GUIDE PART 1 - INTRODUCTION

This brief Part:

(a) provides an overview of the procurement process for new construction ships

(b) identifies the need for consistency in the specification processes

(c) defines the credentials of Guide users

(d) states that the Guide must be followed to generate specification, tenders and evaluation reports in defined, compatible, structured formats.

8. **GUIDE PART 2 - SPECIFICATION PREPARATION**

An overview of Guide Part 2 is shown in Figure 6.3. Chapter 1 is a "common" introduction.

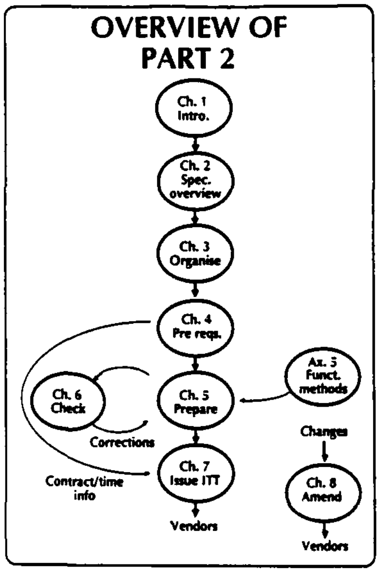


Fig 6.3

Chapter 2 emphasises the importance of an adequate specification because this is the document against which the MCAS system will be developed, implemented, tested, supplied, installed and set-to-work. Perfect specifications will not guarantee perfect solutions, but poor specifications will definitely cause problems.

The Guide then states that the specification must avoid the pitfalls outlined earlier, and in addition must:

(a) state requirements once, clearly and concisely in an easily found location

(b) adopt a functional approach where possible so as to minimise constraints on the vendor's implementation method and encourage innovation

(c) segregate the requirements for the MCAS system facilities from the functional requirements which represent MCAS system application to the machinery systems

(d) be capable of accepting change arising from modifications to the machinery systems.

MCAS SYSTEM SPECIFICATION STRUCTURE

Section	1	Introduction
Section	2	Operational req.
Section	3	MCAS system req.
Section	4	Machinery control and monitoring
Section	5	Typical MCAS system arrangement
Section	6	Design and construction reqs.
Section	7	ARM reqs.
Section	8	Documentation reqs.
Section	9	Training reqs.
Section	10	QA and test reqs.
Section	11	Scope of supply
Appendix	1	Related documents
Appendix	2	Glossary
Appendix	3	Signal schedule
Appendix	4	Signal distribution info.

Fig 6.4

Figure 6.4 summarises the MCAS system specification structure that the Guide imposes.

Note how all the related aspects such as ship description, operational aspects, MCAS system facilities, Functional Definition, Constructional requirements, testing, supply and support requirements etc are all located in separate sections of the same document, so that they may be interpreted and understood in total context.

The Guide contains a "tree diagram" summarising the specification structure and contents at section and sub-section level (Fig 6.5). The contents/scope of this diagram (representing the total MCAS specification contents) has been carefully developed to include all possible aspects that need to be defined for MCAS system specification. Adherence to the Guide should therefore result in an MCAS system that is complete and logically presented, and all MCAS system specifications will have the same structure.

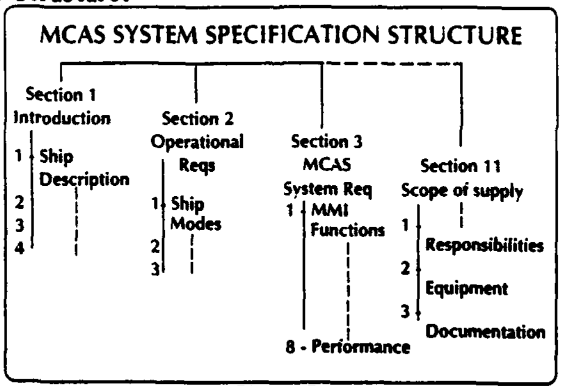


Fig 6.5

8.1 Pre-Requisite Activities

Prior to undertaking preparation of the MCAS system specification it is strongly recommended (Guide Part 2 Chapter 4) that the specifier compiles a database of information which will eventually be expressed as MCAS system requirements. The Guide identifies typical source documents and identifies the type of information that should be gleaned from each, generating a traceable link between the specification requirements and their original sources.

8.2 Specification Details

The Guide then provides (in Chapter 5) detailed guidance on the preparation of MCAS system specification Sections 1-11 as identified in Figure 6.5. To assist with this procedure the pullout "Tree Diagram" which indexes the specification Section/Sub-Section contents has been endorsed with references to the Guide paragraphs where relevant guidance will be found (Figure 6.6).

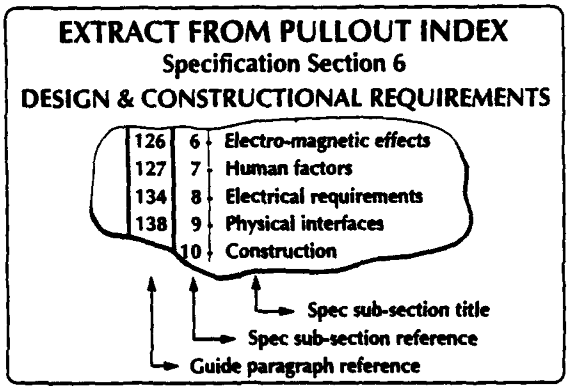


Fig 6.6

Further information on some of the Specification Sections generated by following the Guide is given below.

Specification Section 2 defines how a given number of men will control the machinery in predefined modes from the various operating positions under various states of readiness.

Specification Section 3 defines the requirements for the MCAS system facilities e.g. MMI aspects, Alarm/Warning handling, system modification requirements, interfacing requirements, data update rates etc.

Specification Section 4 has the major task of defining in functional terms how the MCAS system should be applied to control each of the machinery systems (referenced 1, 2....N) in turn. It will address each machinery system in four ways:

(a) to produce a brief description of the machinery system to be controlled

(b) to specify MMI functions that the MCAS system must provide to display information on the machinery systems and to accept operator control input demands.

(c) to specify automatic control functions that the MCAS system must provide for the machinery system including identification/definition of control procedures, algorithms, calculations etc.

(d) to define the interface requirements between the MCAS system, the machinery and the other ship systems.

Fig 6.7 shows how the specification will reference each of these aspects for the various machinery systems.

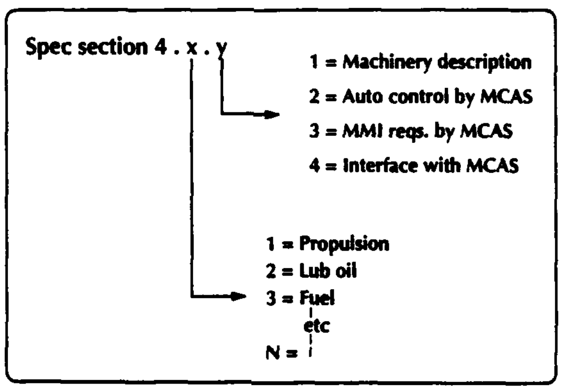


Fig 6.7

The detailed data compiled in these activities will be accommodated in a computerised signal schedule.

A Functional approach (e.g. YOURDON) is recommended to express these requirements. Fig 6.8 shows the MCAS system functional breakdown in the form of a Context Diagram. The functions shown would be expanded by decomposition until the appropriate level of definition had been achieved, and dataflow tables would be developed to define the information flow between functions. Function identifiers "2, 3, 4" for control, MMI and interface functions respectively in the Context Diagram correspond directly with similar identifiers in Fig 6.7 assisting cross-referencing in the documentation. Examples are included in the Guide for illustration.

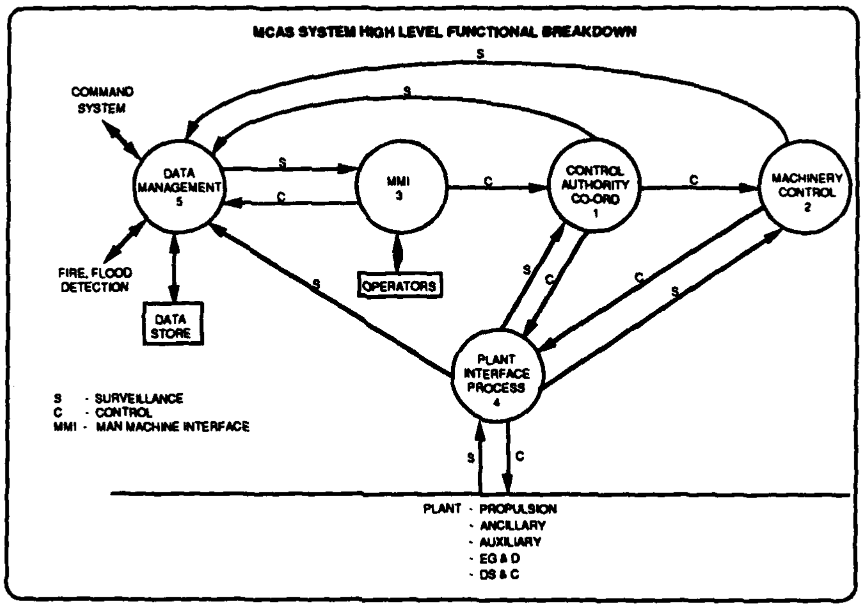


Fig 6.8

It is emphasised that precision and clarity are vital for this important section on functional decomposition which provides an "interface of understanding" between the client's design objectives in the context of ship control, through the development of procedures/algorithms by an engineer, to the preparation of code by a software writer who may have no real knowledge of the application. The Guide identifies levels of definition as benchmarks for testing during the build and supply phases.

Specification Section 6 details all of the constraints on system implementation, ranging from weight, power and size allocations to environmental requirements and from software standards to system vulnerability.

The Guide also specifies that a Human Factors programme should be carried out to ensure a user-friendly MCAS system.

The remaining Specification Sections 7 to 11 clearly specify the reliability/maintenance requirements, identify all system documents (e.g. FMEA) required, (their use and who should generate them), quality and test requirements and the scope of supply. Operation and maintenance training needs are also defined.

8.3 Specification Issue

The Guide advises how the completed specification should be checked, issued as part of an Invitation To Tender (ITT) and if necessary modified/reissued to accommodate update due to machinery system changes. A strict policy of configuration control is recommended to supervise the documentation.

8.4 Levels of Specification

Specifications may be issued at a high or low level. In the first instance a conceptual/high level specification may be issued to test the market for cost, project approach and typical solutions, or to short list a number of potential vendors.

More detailed level specifications would impose further constraints on the vendor, who may be tasked with developing the functional definition aspects to a lower level definition thus demonstrating his capabilities, and providing a more firm basis for system costing.

In the extreme, issue of a totally complete/defined MCAS specification would represent a "build" specification for which the vendor would have no functional responsibility.

Irrespective of the level of specification, the concept of a fixed format and the use of a suitable word processor package will do much to simplify document update, reissue and configuration control.

9. GUIDE PART 3 - TENDER PREPARATION

At the outset the vendors, who have received the Invitation To Tender (ITT) are encouraged by the Guide to read it in detail and ensure that all the requirements are fully understood. Any queries should be addressed to the specifier for clarification. Sufficient time should be allowed for this important stage prior to preparing a tender response.

Vendors are advised:

(a) to check their resources and make a decision on whether or not to bid

(b) that they will be totally responsible for any sub-contractors

(c) that their tenders will only be evaluated on the evidence contained within them

(d) that a team of specialists should be formed and co-ordinated to generate the tender

(e) that tenders should be prepared in a predefined format as described below.

9.1 Tender Format

Figure 9.1 summarises the contents/structure of the tender document.

TENDER FORMAT

Section	1	Introduction
Section	2	*Operational aspects*
Section	3	General MCAS system aspects
Section	4	Application of MCAS system to machinery control
Section	5	Overview of proposed system structure
Section	6	Design and constructional features
Section	7	ARM capability
Section	8	Documentation
Section	9	Training facilities
Section	10	Provision for QA, test, installation & support
Section	11	Items to be supplied
Section	12	Summary of compliance
Section	13	Company information
Section	14	Timescale/contractual information

Fig 9.1

Sections 1 to 11 of the tender are intended to reflect the requirements of MCAS system Specification Sections 1 to 11 respectively.

Tender Section 12 will contain statements/tables identifying compliance with the specification. Any non-compliances (which are not necessarily signs of weakness) must be fully justified.

Tender Section 13 should include:

(a) supporting information on the vendors/sub-contractors track record, resources and financial standing

(b) CVs for all key personnel with assurances of their availability during the project

(c) assurance of technology availability as proposed for the MCAS system solution

(d) information on quality procedures/standards to be adopted.

Section 14 of the tender should show the vendor's proposed timescales for receipt of information, progress meetings, build and test phases, deliverables and commissioning. Any long lead items should be clearly identified.

Part 3 of the Guide also contains a pull out Tree Index for the tender document, identifying the Section headings and contents.

Guidance is also given on checking the tender before issue to ensure that it correctly reflects the customer's requirements.

It will be seen that by following the predefined tender structure and completing the compliancy statements the vendor has been forced to produce a tender which is complete, reflects the specification structure and facilitates the evaluation process.

10. **GUIDE PART 4 - TENDER EVALUATION**

Guide Part 4 is directed at the Evaluator who will systematically compare the tender against the original specification requirements and determine/recommend the most cost effective MCAS system solution.

At the outset the Evaluator is required to organise a team of specialists who will read and fully understand the MCAS system specification and the tenders received. Queries on any of these documents should be immediately clarified so that the evaluator may form his judgements on known facts.

The evaluator will be advised:

(a) that all assessment must be based solely on the tenders received

(b) that specialists must be assigned to assess particular aspects of the tenders to maintain consistent subjectivity of judgement

(c) that all decisions must be recorded.

A four stage evaluation process is proposed:

(a) the preparation of questions, based on the specification

(b) award of marks for compliance for each of the questions

(c) numeric analysis to 'normalise' the results

(d) the presentation of results in an easily assimilated format.

10.1 Preparation of Questions

The team of evaluation experts are guided to generate a questionnaire which will be used to interrogate the tenders against the specification.

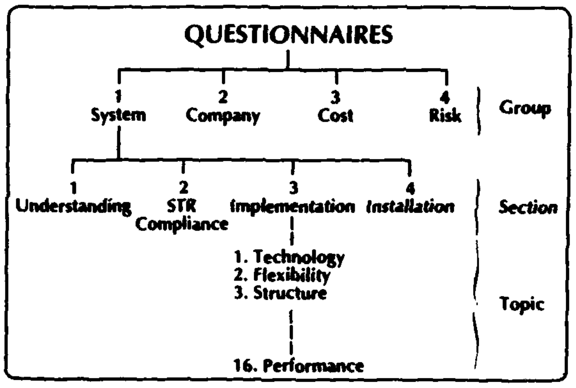


Fig 10.1

Figure 10.1 summarises part of the evaluation questionnaire and shows how high level aspects can be sub-divided down to a Topic level at which specific detailed questions may be asked.

The scope of the evaluation questionnaire can be broken down into four main groups:

(a) SYSTEM aspects - which assess the technical content of the proposed solution

(b) COMPANY aspects which assess the general capability, resources and attitude of the vendor to provide the solution

(c) COST aspects which assess the attention given to providing a realistic cost estimate (UPC/TLC)

(d) RISK aspects which highlight any aspects of the proposed solution which could prejudice implementation due to technical or contractual reasons

The questions should be compiled on a requirement-by-requirement basis using the specification/ITT package for reference.

Pro-forma worksheets are supplied to:

(a) identify the evaluator

(b) identify the project reference

(c) identify each question by a code comprising group/section/topic/question number

(d) record the question

(e) identify the specification/ITT statement against which the question was generated

(f) record the mark and weighting factors as defined below.

10.2 Marking, Analysis and Presentation

Marks for levels of response compliancy may be typically awarded in the range 0 to 5 where

5 = outstanding - beyond requirements

4 = totally acceptable - exceeds some requirements

3 = satisfies minimum requirements

2 = needs improvement - contains minor deficiencies

1 = fails to satisfy requirements - unacceptable

0 = not addressed.

Cost information should be assessed separately to technical information to maintain objectivity of judgement.

Numeric Analysis of the basic marks awarded against each question is required to:

(a) cater for variations between the number of questions set against each topic

(b) cater for the relative importance of some MCAS systems aspects compared with others

(c) obtain overall figures of merit at subject, section and group levels

(d) obtain an overall relative score for the systems assessed with reference to the same baseline

(e) provide a basis for the concise display of results.

Specialists other than those responsible for the basic marking should determine weighting factors in the range 0 to 1 (where 1 has the highest priority) for each topic/section/group and apply them to rationalise the scores (as percentages) at topic, section, group and overall levels e.g.

$$\text{Group Mark} = \frac{\sum(\text{Section Marks x Section Weighting Factor})\text{x}20}{\sum(\text{Section Weighting Factors})}$$

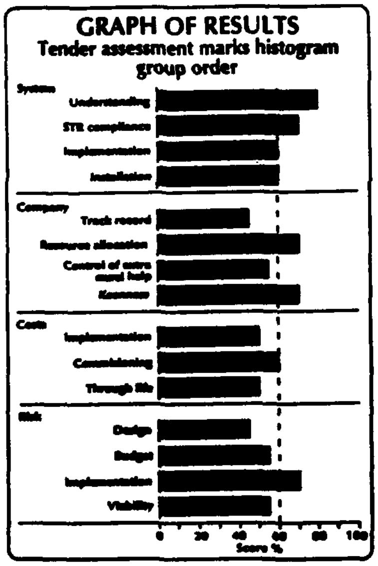


Fig 10.2

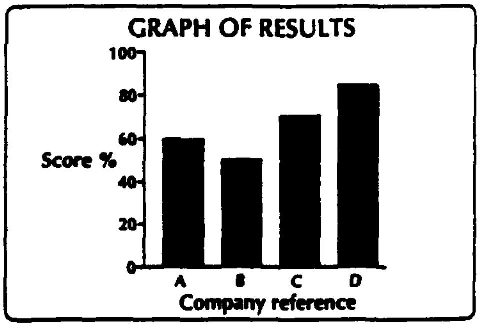


Fig 10.3

The scores may be presented in histogram form as shown in Figures 10.2 and 10.3 which readily identify a single vendor's scores for each aspect of his tender, and the overall scores for various tenders.

Note that for confidentiality, coded references are assigned to conceal vendor identities.

Adherence to the Guide therefore ensures that the evaluation process is fully recorded, is fair and complete.

11 GUIDE PART 5 - SUPPORTING INFORMATION

It is not an objective of the SSCP27 MCAS System Specification Guide to re-state supporting information that is readily available in standards, textbooks or well publicised technical documents.

If such information has been identified to support some statement anywhere in the Guide, then adequate cross references are included in the text.

However there are certain topics which are only encountered on specialist applications such as warship control, of which a newcomer to the field of naval MCAS systems may be unaware, e.g. machinery system interdependencies, citadels, zoning, vulnerability, damage control, manning and responsibilities etc.

There are also instances where 'well known' subjects such as reliability must be interpreted in the context of MCAS system application e.g. availability in terms of Bridge control if the SCC facilities fail.

Part 5 provides a digest of information on topics including:

(a) Ship and machinery systems

(b) Damage Control

(c) Operational Aspects

(d) MCAS System Aspects

(e) Methods of Functional Definition

(f) Software

(g) Availability, Reliability and Maintainability

(h) Through Life Costing, Upkeep and Support

in the context of a naval MCAS environment.

12. **CLOSING REMARKS**

The Guide will provide a much needed structure and assistance in future MCAS System Specification.

At the time of writing SSCP27 has been completed to include MOD(PE) comment and is about to be exposed to those involved in the MCAS industry to obtain "user" comment.

Following the incorporation of user comment it is expected that SSCP27 will be formally issued by MOD(PE) in the second half of 1990 for use during the procurement of future MCAS systems or for major modification to existing MCAS systems.

It will be readily appreciated that the concept and principles contained within the Guide are also relevant to complex, software based systems in general. Investigations are underway to explore other areas of application.

13. **REFERENCES**

[1] T.McClean - "The Specification and Procurement of Complex Systems"
Colloquium on Specification of Complex Systems, Digest 1989/145, Institution of Electrical Engineers, London, November 1989.

14. **ACKNOWLEDGEMENTS AND DISCLAIMER**

This work has been carried out with the support of the Procurement Executive, Ministry of Defence.

The views expressed in this paper are those of the authors and do not necessarily represent the official views of the Ministry of Defence.

DEVELOPMENT OF A 5000 POINT CONTROL AND MONITORING SYSTEM

by C T Marwood
and M C Glover

1. ABSTRACT

Control and monitoring of platform equipment on the first of class of a new fleet support vessel will be carried out by an extensive system with multiple control locations using VDU terminals. The development of this system has involved special techniques for definition and build to meet requirements which go beyond the average for this type of ship.

The usefulness of these techniques and scope for future extension are discussed.

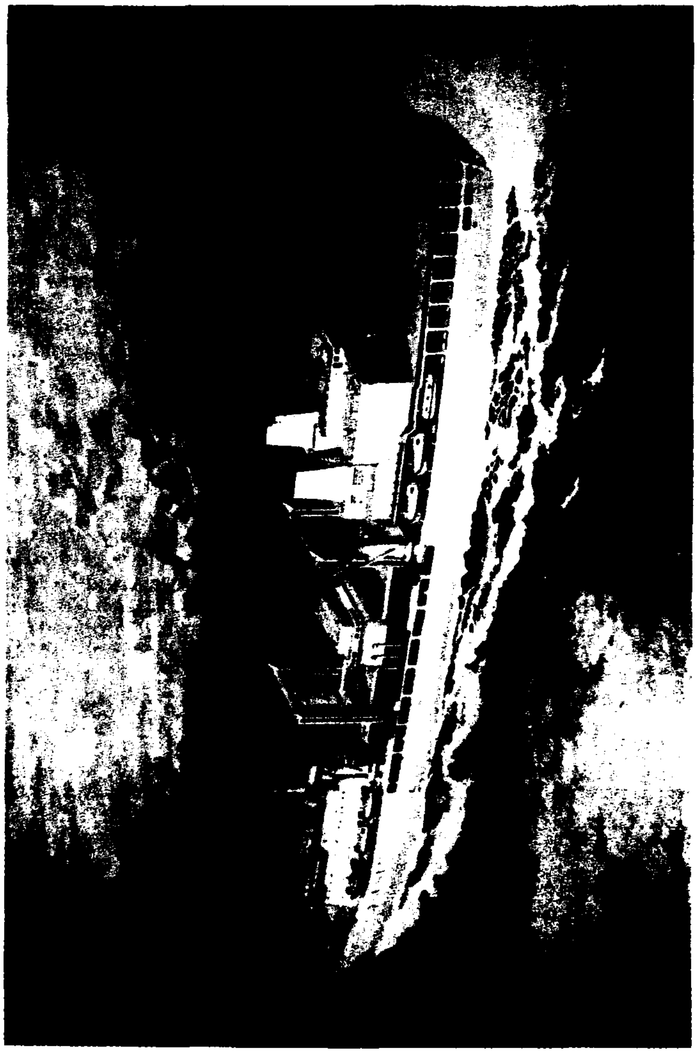
2. INTRODUCTION

The scope and functionality of the Machinery Control and Surveillance System (MCAS) for this ship (Fig 1) represents a major step forward. This arose from the combined effects of the wide range of equipment for 'one-stop' replenishment, and pressure to reduce manning levels. The main features of the system were already established by the UK Ministry of Defence in conjunction with Harland and Wolff at the time that the MCAS contract was let. Further details are given in Ref (1).

2.1 Equipment

The equipment to be controlled and monitored includes:

Propulsion	Twin Shafts with medium speed reversing diesels, clutch and brake
Ancillaries	Fuel, lub oil, cooling, start air compressors
Electrical	Six diesel generators with three 3.3kV switchboards and low voltage distribution
Cargo/Ballast	Approximately 70 tanks, 25 pumps and 200 valves



Damage Control	Fire pumps and valves, ventilation clearance fans, sprinklers, foam, halon, plus flooding, door and hatch indication.
Auxiliaries	Refrigeration, chilled water, boilers, compressed air and domestic services.

Replenishment at sea calls for simultaneous operation of most of this equipment, with Damage Control Systems ready for use in case of emergencies.

The number of operators for these tasks must be kept to a minimum, although the amount of machinery is higher than in some earlier ships. This requires increased use of remote control and surveillance, plus automation of some functions which were previously carried out manually. For example, sequence re-starting of essential services and motor drives following electrical black-outs to avoid overloading the generators.

The number of control and monitoring signals shown in Fig 2 are necessary for the present equipment. In addition, spare capacity has been included to allow for future growth.

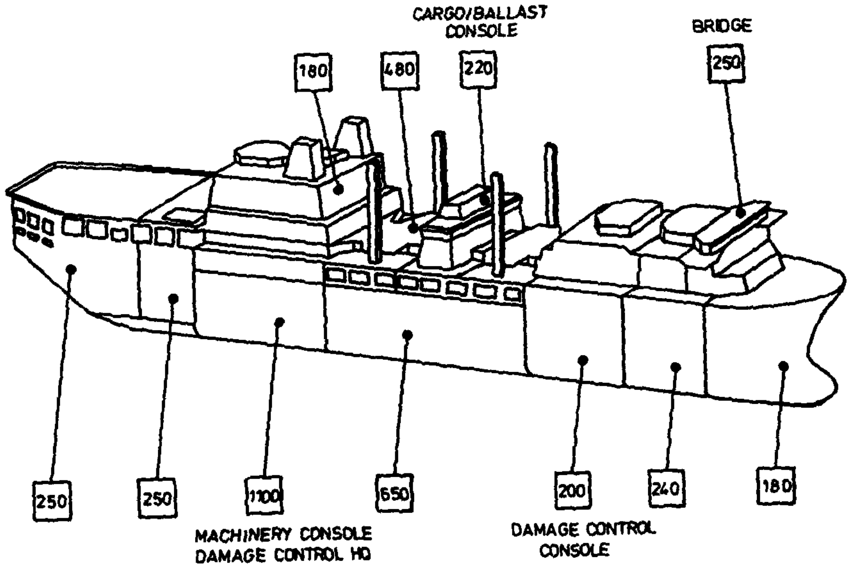


FIG 2. OPERATING POSITIONS & SIGNAL DISTRIBUTION

2.2 Control Positions

The requirement for availability after partial damage and the need to split the workload between multiple operators called for a number of remote control positions located at various parts of the ship, some functions available at more than one location. For example, damage control can be carried out from the main headquarters in the machinery control room, from the secondary headquarters or from the Bridge. All main machinery and equipment is fitted with local backup control independent of MCAS.

3. SYSTEM STRUCTURE

The initial design considerations which determined the system structure have been described in Reference (1). They include:

(a) Selection of a fully distributed system, ie one with no centralised co-ordination of control functions, rather than other combinations such as centralised control and distributed monitoring. This was chosen to avoid dependance on central processors and to enable a degree of stand-alone operation in the distributed units. (2)

(b) Combining control and monitoring capabilities in each of the distributed processors, known as Outstations, while ensuring the independence of safety, control and alarm functions required by Classification society rules.

(c) Control of equipment from VDU terminals with keyboards and mimic pages, combined with dedicated panels for rapid operation o emergency trips and for backup control. This reduces th [illegible]e of [illegible]rol consoles and enables all alarms and equip[illegible] [illegible] o be displayed at any of the operating positio[illegible] [illegible]ntrol and alarm acceptance are assignable to specific positions.

(d) Lever control is retained for propulsion at the Bridge manoeuvring console, Bridge wings or machinery control room.

(e) Dual redundant central stations and display processors.

(f) Dual redundant communications network linking the central stations to outstations.

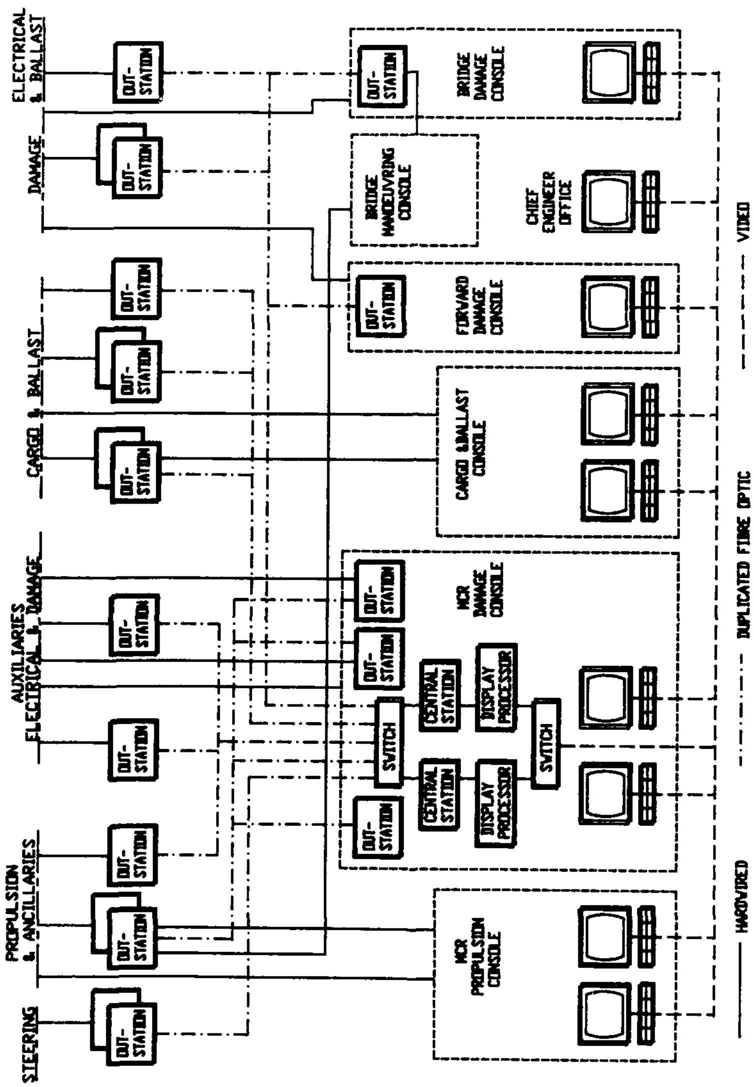


FIG 3. OVERALL SYSTEM

(g) Use of existing software and hardware to minimise development cost, which was one of the main selection criteria for a fixed price contract.

The system structure which resulted (Fig 3) was developed jointly with Harland and Wolff and their consultants Shell Seatex. 20 Outstations are spread throughout the ships, 15 in locations chosen to minimise cabling from machinery and 5 mounted in consoles to connect with switches and indicators. 4 basic types of module are used in all outstations; processor, scan controller, analogue and digital interfaces. Up to 8 interface modules in any mix can be fitted, giving a capacity of 256 channels. All channels are accessed by the scan controller, which transfers data to memory for the processor module, which controls all other functions are provides communications.

Outstations are linked to both central stations by a duplicated fibre optic network, described in detail later. Central stations also use the standard range of modules, but have no machinery interfaces. They control the communications network and pass data to the Display processors, which drive the console mounted terminals. All VDU/Keyboard terminals can be driven by either central station, and are used for control as well as monitoring and alarms. Dedicated control panels at the Bridge, Forward Damage and Headquarters Damage consoles are connected to the data network, using console-mounted Outstations. Facilities available at each control position are shown in Table 1.

Directly wired connections provide remote control and monitoring at each of the consoles totally independent of the central stations and data network. (Fig 4). Some simple controls such as trips are wired from panels to machinery. Others, like the propulsion levers, connect via an Outstation which provides co-ordination and interlock protection.

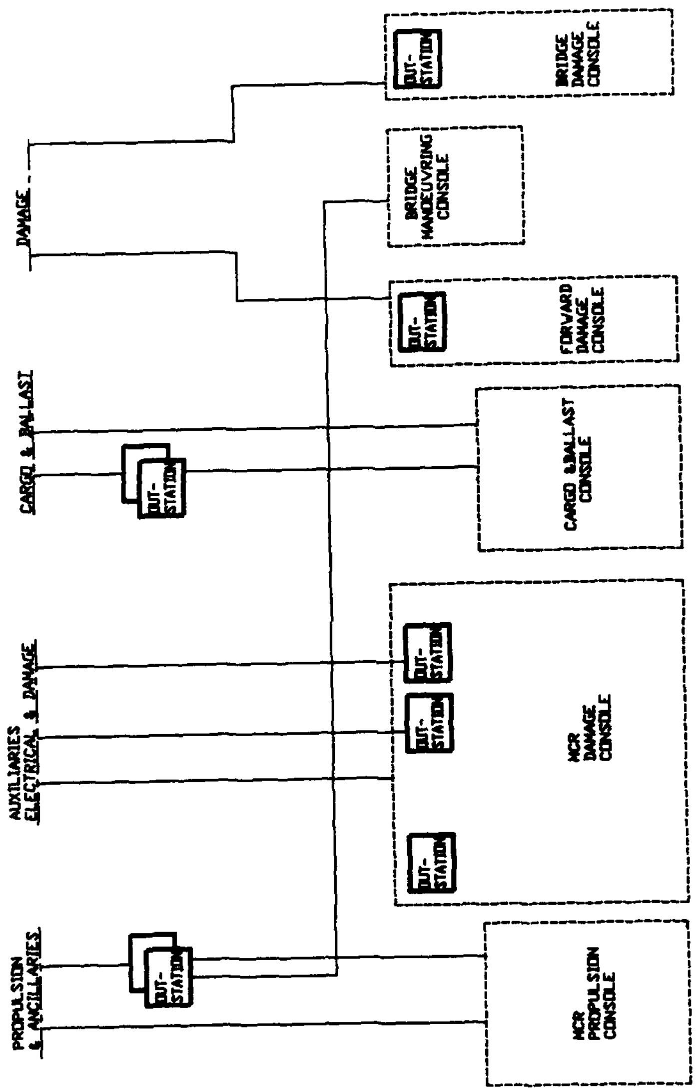


FIG 4. HARDWIRED CONNECTIONS

<table>
<tr><th>WORK STATIONS</th><th>MONITORING</th><th>CONTROLLING</th><th>ALARM ANNUNCIATION</th><th>ALARM ACKNOWLEDGE</th></tr>
<tr><td>MCR Propulsion Control Work Stations</td><td colspan="4">Propulsion and Auxiliary</td></tr>
<tr><td>Main RAS Work Stations</td><td colspan="4">Replenishment at Sea (RAS)</td></tr>
<tr><td>MCR Damage Control Work Stations</td><td>Damage, RAS</td><td>Damage</td><td colspan="2">Damage, RAS</td></tr>
<tr><td>Bridge and Forward Damage Work Stations</td><td>Damage, Propulsion and Auxiliary</td><td colspan="3">Damage</td></tr>
<tr><td>Chief Engineers Office/portable</td><td>Propulsion and Auxiliary</td><td colspan="3"></td></tr>
</table>

TABLE 1 OPERATING FACILITIES

4. DEFINITION

4.1 Functions

Early in the project a formal method was chosen for specifying functions to be performed by the system. The main objectives were to avoid ambiguities inherent in written specifications, and to enable automatic code generation to be used. Written specifications have been adequate up till now, but with the large number of control and monitoring tasks to be specified, some using remote or automatic control for the first time in this type of vessel, a better method was needed. After evaluating a number of software tools, a formal method using flow diagrams was adopted, which significantly changed the control definition procedure.

For example, a simple pump control could be specified:

1. When STOP is selected, the pump will stop
2. When RUN is selected, the pump will run continuously
3. When AUTO is selected, the pump will run if main pressure is low.
4. As a general requirement, the control system must fail set, without changing the running state of the pump.

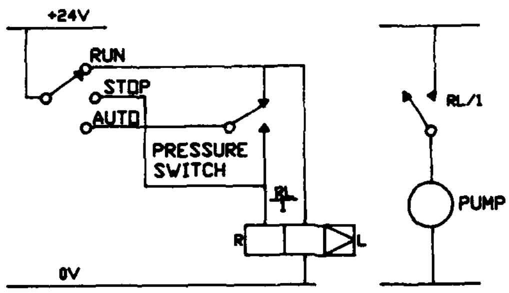
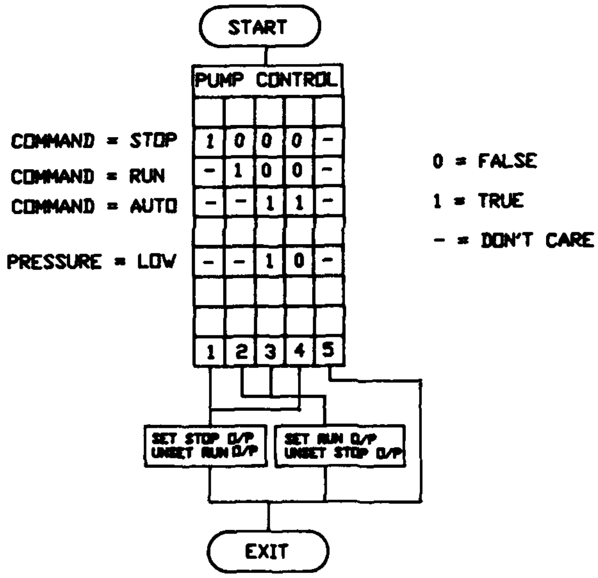


FIG 5. CONVENTIONAL PUMP CONTROL



COMMAND = STOP	1	0	0	0	-
COMMAND = RUN	-	1	0	0	-
COMMAND = AUTO	-	-	1	1	-
PRESSURE = LOW	-	-	1	0	-
	1	2	3	4	5

FIG 6. FLOW DIAGRAM

The specification writer would probably have the circuit of Fig 5 in mind, with a latched relay to ensure Item 4 is met.

Using the flow diagram method, a top level written specification is followed by the flow diagram in Fig 6, with a decision table consisting of a number of columns to be tested one at a time, starting at the left hand column. If all the conditions in a column are true, testing stops and the task linked to the line below the column is actioned. Otherwise the next column is tested. If none of the columns is satisfied, the path from the last one is taken as a default option. An important part of the specification process is to examine the default possibilities and ensure that unsafe combinations of events are properly dealt with. In Fig 6 the default column includes the cases when none of the commands is selected, which the written specification did not cover.

The decision table also defines another area which the written specification omitted; what action to take if more than one command is true at the same time. In first column STOP overrides RUN and AUTO, and in the second column RUN also overrides AUTO. In Fig 5 faults causing more than one command are unlikely, but with multiple control positions it is safer to allow for the possibilities.

The benefits of this method include:

- A thorough specification of the control requirement, which is difficult to achieve in written form, or by conventional flow diagrams. This can be checked automatically for completeness.

- Consideration of error conditions at the definition stage, rather than during detail design and test

- Flow diagrams are easier to interpret and are less ambiguous than text, particularly for complex control.

- Diagrams form effective documentation, and can be used in test and verification.

AOR includes over 50 types of algorithm, with an average of four flow diagrams each. Multiple machinery items such as valves use copies of the same algorithm. Decision tables involved more work at the early stages of development, but saved considerable time and effort in programming and integration.

4.2 Input/Output Database

Definition of all the signals linking the MCAS to machinery and equipment is a task of similar size to function definition when 4000 points are involved, each with over 100 fields of related data. It was essential to work with a single database with shared access to avoid errors and permit automatic checking, so the format and updating of shared areas were agreed with the shipbuilder early in the development.

The input/output database is only one of several linked databases shown in Fig 7. Its fields are split into two groups, entered by the Shipbuilder and MCAS supplier, linked by a common tagnumber. The Applications database contains the software which operates directly on the input and output signals, and generates derived data such as "standby pump started" which can be used to generate internal system alarms for display and control. A simple example of applications software is the pump control algorithm of Fig 6. If there were several pumps with identical control functions, copies of the same algorithm would be used, using different tagnumbers for each pump. The Display database defines the way each signal is presented, and allocates signals to mimic pages.

All this data can be defined and cross checked without allocating tagnumbers to physical channels, or deciding which outstation an algorithm will actually run in. Flexibility in setting up the configuration data can thus be maintained until well into software integration. When the software is to be downloaded into the outstations, configuration files are set up automatically. Consistency between the linked databases is essential, and a database utility was developed for cross-checking and to update associated data automatically if a change is made to one of the databases.

4.3 Allocation of signals and functions

The decisions on outstation locations and signal allocation were made by the Shipbuilder, mainly on the basis of minimum cabling. For major systems and essential equipment the effects of outstation failure were carefully considered, for example separation of port and starboard propulsion. Shipwide control of high pressure seawater pumps and ventilation equipment were distributed between several outstations to ensure that partial automatic control could be retained, in the event of damage. Controls of steering pumps were also split between two outstations.

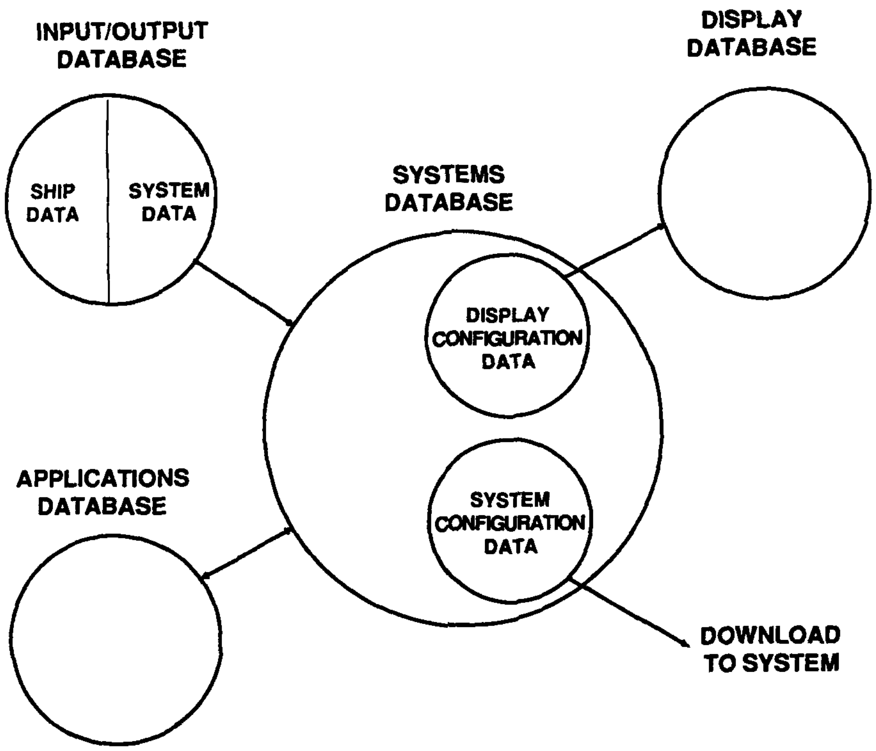


FIG 7. DATABASE STRUCTURE

5. IMPLEMENTATION

5.1 Data Flow

The communications system was designed with the aims of immunity from single faults, tolerance of transient interference, a very high degree of error detection and recovery, and consistent performance in message overload situations. To achieve consistent performance a relatively simple deterministic approach was adopted, using the central station to poll each outstation, with a send-all response. This apparently uneconomic use of link capacity ensures that any event which is not transmitted during one polling cycle,

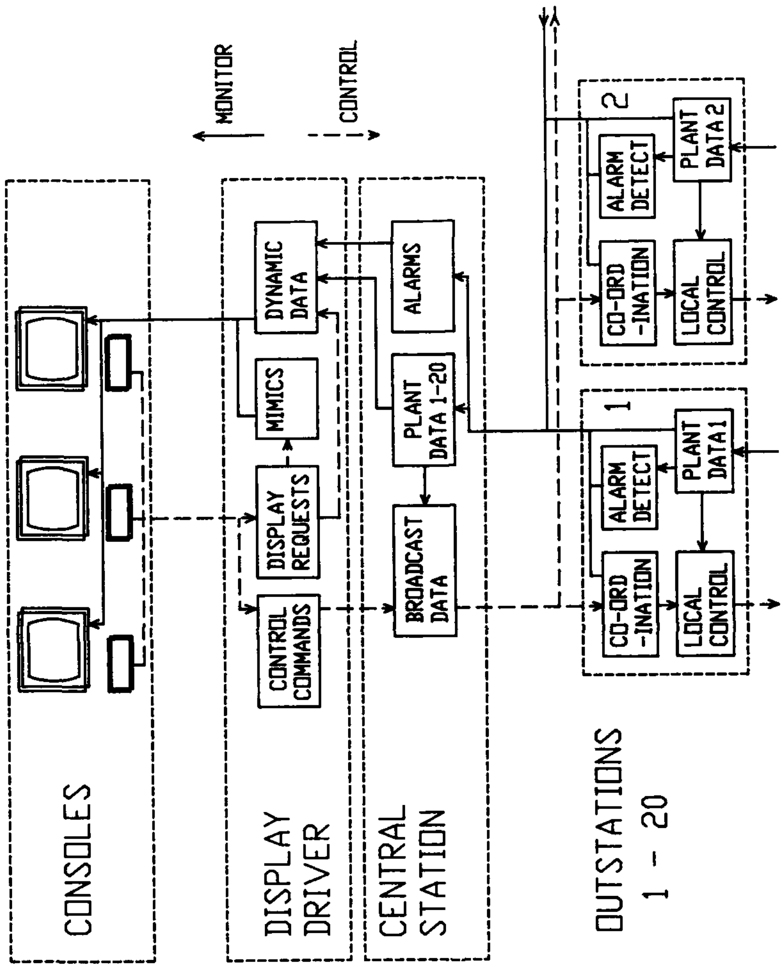


FIG 8. DATA FLOW

for whatever reason, is picked up at the next without complicated reversionary procedures. It also means that if multiple alarms are caused by battle damage or a major fault, control commands can be sent with no increase in delay.

In Fig 8, each Outstation continuously scans plant signals between polls, conditions and validates the inputs and detects alarms. When polled, it sends plant parameter values and derived data including alarms and parameter validity status as well as computed conditions such as 'engine start failed'. Plant and derived data from all outstations is collated in the central station database, where it can be accessed by the display processor. A subset of this database, containing all information for transfer between outstations, is broadcast at the end of the polling cycle, along with plant control commands from the display processor. In simple cases the outstation to which a command is addressed carries it out directly.

More complex control functions are executed in two stages, the first outstation interpreting the functions and passing on simple commands to others. For example, sequential starting of standby pumps is performed by a high level sequence which sends demands to the standard plant interface start/stop algorithm.

In the Display processor, requests for a new page bring up a static mimic supplemented with dynamic information from plant signals and derived data.The main display processor regularly requests relevant data from the central station database, and sends it plant control commands from the keyboards as a higher priority.

5.2 Fibre Optic Data Network

Fibre optics not only gave high data rates but avoided the cost of filtering signals from areas exposed to high levels of electromagnetic pulse energy.

The network structure and components were chosen to satisfy the requirements for a high degree of single fault tolerance, high reliability and to minimise the loss of communication if multiple faults occur. In a ship of this size cable runs of several hundred metres are not unusual, so optical signal budgets to calculate the attenuation of cables and connectors needed special consideration.

a. Structure The 20 outstations are divided into 5 groups, two of which are detailed in Fig 9. Each group of outstations is connected to two star centres by two independent multi-drop chains made up of fibre optic links. The star centres are linked to both central stations also using fibre

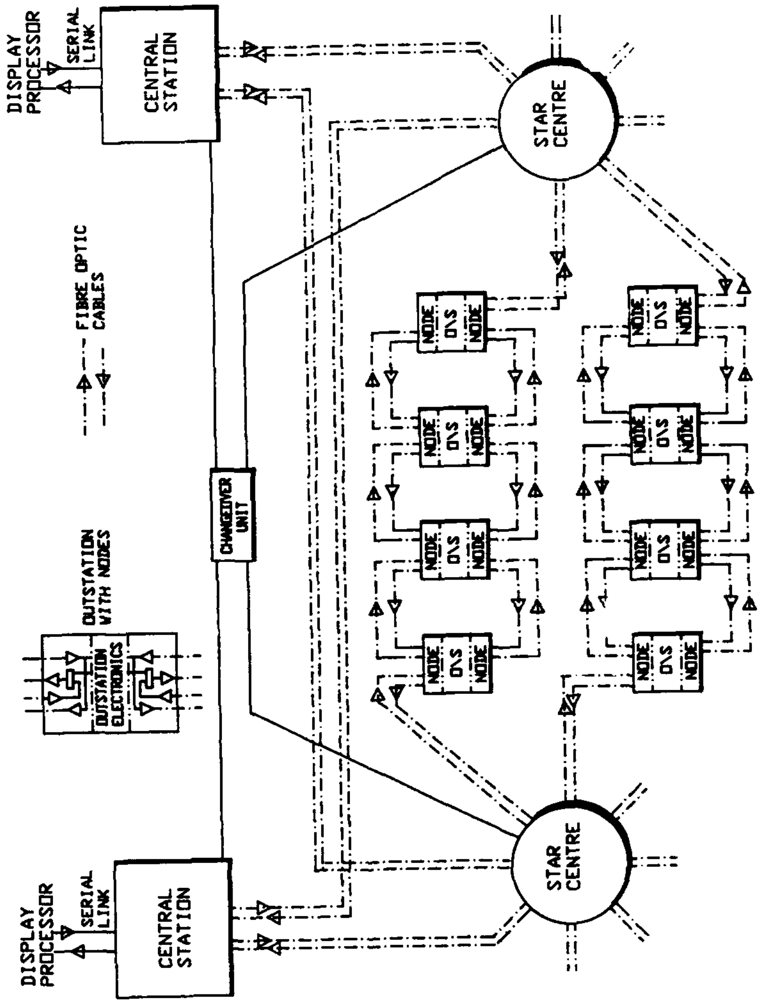


FIG 9. NETWORK STRUCTURE

optic links, with automatic change over switching so that the active central station can use both stars. Each link has separate cores for transmit and receive signals. Two node units in each outstation provide optical/electrical interfaces and regenerate optical signal levels. The two chains for a group of outstations are connected to separate star centres. The connections are arranged so that the first outstation in one chain is the last in the other chain. This ensures that no single failure of a link, node or star centre can isolate any outstation. If one outstation fails, all others remain connected.

b. Nodes. Standard nodes in each outstation and central station interface between electrical and optical signals. The electronics are totally enclosed in a metal shield to prevent electromagnetic interference. High efficiency light emitting diode transmitters were chosen rather than lasers to increase reliability, which is further improved by reducing operating power levels.

Any multi-drop or bus network is vulnerable to faults which cause a processor to transmit continuously, blocking communication between the others. Safety circuits in each node can override the software to limit this condition. They also detect transmission failure.

c. Data Rates. Links operate at 250Kbaud, well within the capacity of the cable. Cable lengths can be up to 250m under worst case conditions including allowance for two extra connectors for repairs. Fitting connectors can be carried out with a simple kit and does not need high skill levels. Practical tests have shown fault-free operation with 1000m cables.

5.3 Fibre Optic Links to Terminals

To ensure that electromagnetic pulse transients could not be conducted down the video and keyboard links from consoles outside the main hull, fibre optic connections were also used for these signals. Change over relays switch signals from either display processor to all terminals, controlled by the Master Station change over unit. This monitors healthy signals from both central station watchdogs, and also switches over star centre links to the active station. Local manual switches can be used to over ride automatic selection if necessary.

5.4 Software Generation

Software for MCAS falls into two main categories, System and Application. The system software performs the general functions shown in Fig 10, many areas of which are the same for any type of ship. The Operating System and Display software are mature proprietary packages, chosen to reduce development risk and cost. Over 600 users of the Operating System include real-time control of electrical generation and railway signalling. The Display software has a history of 5000 applications over 9 years, many with safety critical aspects. It would take considerable effort and cost to devise a validation method equivalent to this extent of usage, so instead the system testing concentrates on the more critical operational aspects.

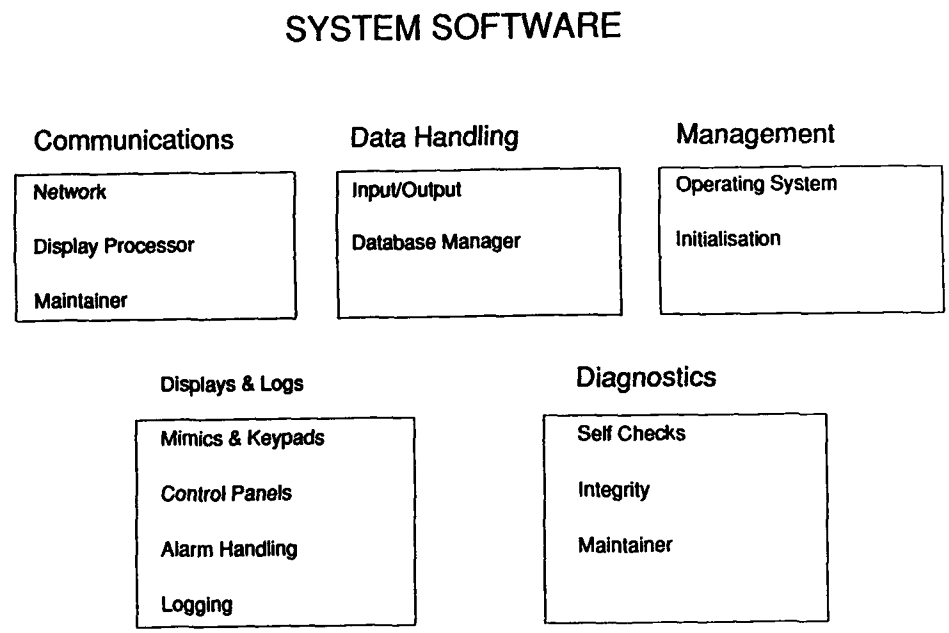


FIG 10. SYSTEM SOFTWARE

In-house system software is mainly in PASCAL, with some assembler for efficient operation, and is designed to reduce the Applications software workload by providing a framework including data routing, start-up initialisation and fault handling.

Application software is configured specifically for each ship's control and surveillance requirements (Fig 11). All the function algorithms were defined using the flowchart method described earlier.

Conventional development cycles for projects of this type start with a System Design specification, defining total functional requirements for both hardware and software. Below this, the Software Functional specification defines the structure required to meet the functionality, and accurately specifies interfaces between systems and application software. Four levels of documentation follow:

- Level 1 breaks the system down into a number of functional facilities and major software tasks.

- Level 2 gives a further breakdown into modules, and adds detailed data descriptions.

- Level 3 contains detailed descriptions of each module.

- Level 4 consists of actual program listings.

Using the new method for Applications software, the System and Software Functional specifications are kept, but the flow diagrams themselves contain sufficient detail that the content of the Level 1 specification is reduced and there is no need for Levels 2 and 3. Costs and timescales are reduced, but most significantly changes to requirements can be handled quicker and with less risk of errors.

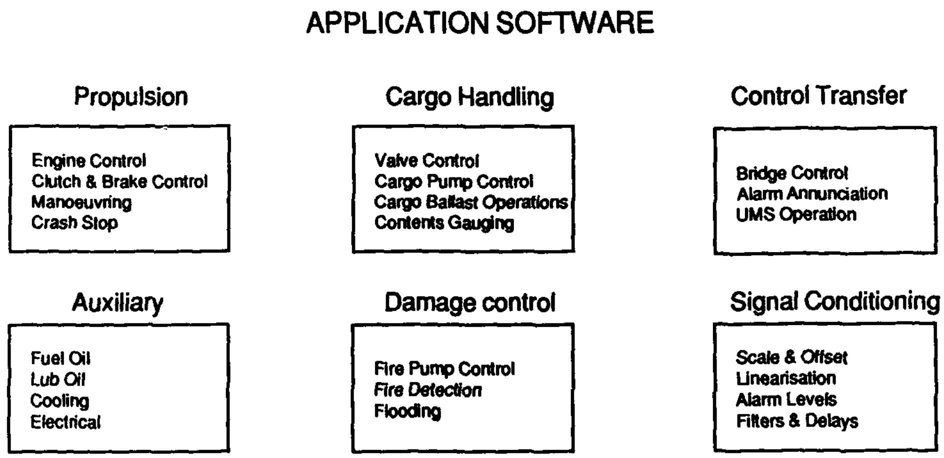


FIG 11. APPLICATION SOFTWARE

5.5 Outstation Firmware

Application and system software are held in each outstation and central station in memory on the processor module. All code and data to allow the system to self check, initialise and begin operation are held in non-volatile memory, so no downloading is necessary at power up. The simplest implementation would be to configure each processor with only sufficient memory for its specific tasks; however this would make each of the 20 outstation processors unique, and ship spares would need to include at least one of each type. Instead, a decision was made to make all outstation processor modules interchangeable, with sufficient EPROM for software common to all outstations, and non-volatile EEPROM configured with specific data for each. A copy of EEPROM data is held on the adjacent scan controller module. If a processor fails and is replaced by a common spare, the EEPROM is copied automatically from the adjacent module during initialisation.

Sumchecks on both modules are used to test for corruption.

5.6 Fault Detection and Diagnostics

It was recognised from the outset that a system of this size and complexity would need significantly better techniques and equipment to ensure fault detection and fast repair times. The electronics is widely distributed, and no increase in average maintainer skills can be assumed.

From experience of 40 digital warship propulsion systems, fault detection and diagnostics were targeted at four main needs:

- Fault detection as close as possible to the source, to avoid bad data circulating within MCAS.
- Primary indication as to whether the fault is within MCAS or outside it, since sensor failures are expected to be one of the major categories.
- Avoiding unnecessary probing of electronics or module changes in healthy equipment.
- User friendly displays and diagnostic procedures, with values in engineering units and text messages.

a. Fault Detection. Fault checks on processors and memory are carried out during initialisation and, together with other checks, as a continuous online background activity. Fault handling software categorises each fault and takes appropriate action. Where safety may be at risk if processing is allowed to continue, an error code is set and an immediate interrupt is made, freezing all control action. If there is time to prepare for shutdown, as when failure is detected, fault information is stored and an alarm given before control processing is halted, leaving diagnostics running. The remaining outstations and central stations recognise the fault status or inability to communicate, and try to maintain as many functions as possible. Lower fault categories do not require processor shutdown, and the action depends on the type of fault. Communications failures cause a retry procedure before changing over to the standby network, while sensor faults may cause a control function to be suspended, or only result in a warning if the signal is for monitoring.

b. Fault Indication. All detected error codes are shown on a two-digit display visible on a module panel. Whenever it is safe to allow processing to continue, error information is sent to the central station for display at operator consoles.

Operators can take any immediate action necessary, and notify the maintenance team to test the outstation. If a communications fault prevents this reporting method, alarms are indicated locally and a hardwired group alarm signal is sent to the machinery control room. Further investigation and diagnostics are carried out using the Maintainer Unit.

c. Maintainer Unit. The need for checking out the electronics using clear displays and simple operating procedures called for a diagnostic unit with a range of facilities. Fault indications, continuous updates of signal values, help information and alarm acceptance facilities are needed to speed up localisation and repair of the failed item. If these facilities were provided in each outstation, significantly more memory would be needed and the diagnostics facility could be damaged by equipment faults.

Instead, a rugged portable IBM PC-compatible processor provides display and memory, and connects to outstations or central stations by a serial link. The host software need only contain a gateway to allow the Maintainer Unit to read memory, and, with access protection, to write to specific areas.

6) CONCLUSIONS

6.1 Development Solutions

A summary of the main requirements and solutions is given in Table 2. The theme of fault tolerance and robustness has been mentioned in several areas, reflecting an underlying need in a system of this size for the effects of any single fault to be limited. This need has been met by duplicating the key elements of the system, and providing backups for essential tasks. The wide range of remote and automatic control functions were specified and programmed using a software tool based on flow diagrams, which has proven to be very thorough. Expressing requirements at this level of detail and learning skilful use of the method required extra effort, but these were offset by reduced documentation and a better definition to test against.

Fibre optics have exceeded expectations in providing high bandwidth at comparatively low cost, with the added bonus of reducing filtering against EMP. With improved tools and techniques now available, installation and repair skills can be learned quickly.

A shared database is essential - apart from the initial work to set it up, the effort to update and maintain it should not be underestimated.

Operating system software needed no modifications, but shipboard procedures called for some further extensions to the wide range of facilities offered by the Display System. Compared with the cost of developing either, the benefits of off-the-shelf software have been considerable. Cost savings from automatic code generation have come mainly from documentation, which is a high proportion of development effort.

REQUIREMENT	SOLUTIONS
Fault tolerance	Duplicated electronics and network Redundant data flow Hardwired back-up
Complex functionality	Flow diagram definition Configurable hardware Flexible algorithm allocation
Fast communications with electrical isolation	Fibre-optic network
5000 Input/output points	Shared database 256-point outstations Automatic downloading
Low-cost development	Proprietary software Modular hardware Automatic code generation
Low repair time	Maintainer unit Standard processor module

TABLE 2 SOLUTIONS SUMMARY

Improved displays on the Maintainer Unit have already proven useful during test, and keyboard operation generally presents no problems to younger engineers brought up on home computers. This solution opens the door to a wide range of facilities, some of which are discussed below.

6.2 Extension of Techniques

a. Hardware Redundancy Fault tolerance and performance can be further increased by fitting an extra processor in each outstation or central station, using spare slots in the VME bus and an alternative version of the Operating system. Workload

could be shared between processors in normal use, with fallback to either one if a fault occurred. Alternatively, where input/output availability is essential, a complete outstation can be duplicated, with automatic change-over switching for control signals only. Monitoring inputs can be connected to both units, with automatic selection of a valid sensor out of any pair.

b. Software Redundancy In the existing design, each Application software task is held at all outstations, but is only configured to run in one of them. If the broadcast message is extended to include all plant and derived data, an application can be run in any outstation, with on-line selection. If an outstation fails, a copy in an alternative outstation can be activated automatically to take over the function.

c. System Definition Much of the work of defining a new system can be reduced using a catalogue of standard control functions, including sequences, interlocks and selection of standby equipment. This would help the buyer choose the best control methods, and the seller to minimise costs by re-using available software.

d. Diagnostics A Maintainer Unit with processing capability, disk memory and interactive displays can be enhanced during the life of the system to include:

- Analysis of error data to locate failed components.

- Logging of selected parameters.

- Trend displays.

- Automatic logging of maintainer adjustments.

In addition to a portable unit, full data broadcasting would allow one or more diagnostic processors to analyse data offline for MCAS health monitoring.

e. Multiple Displays Display processing facilities can be added to any network node to provide extra monitoring positions or local control facilities. These could be used for damage control in each zone of the ship, and for co-ordinated backup control of machinery if a major part of the network suffers damage.

f. Network communications can be linked to other onboard systems to allow MCAS data to be accessed by offline processors for machinery health monitoring or other purposes. Information

on ship speed, heading and water depth could be received from the navigation system and added to the manoeuvring display. A suitable gateway is needed to interface other systems to protect against corrupted commands or messages. (3)

6.3 Summary

The development of this system has involved the use of a range of techniques, mainly derived from existing practice outside the marine equipment business. Once the learning curve has been overcome, these have proved successful and are capable of further extension.

7. ACKNOWLEDGEMENTS

The authors acknowledge the help of colleagues at Hawker Siddeley Dynamics Engineering Ltd in preparing this paper, and wish to thank Harland & Wolff Plc. This work has been carried out with the support of the Procurement Executive, Ministry of Defence.

Opinions expressed are those of the authors.

REFERENCES

(1) C T Marwood "Development of a Ship-wide Control an Surveillance System for Fleet Replenishment Vessels".

Institute of Marine Engineers, London December 1989

(2) C T Marwood "Benefits of Totally Distributed Control Systems" Maritime Communications and Control, October 1988 p. 11.5

(3) C T Marwood "The Growth of Shipwide Digital Systems" 8th Ship Control Systems Symposium, The Hague 1987 p. 5.14

SHIPBOARD READINESS REPORTING
SYSTEM (SRRS) - LEVELS OF REPORTING

by Stanley J. Connors
Naval Ocean Systems Center

1. ABSTRACT

It is more important now, than ever before, for U. S. Navy surface combatants to be integrated from a readiness assessment and reporting point of view. This is because surface combatants are now necessarily more complex as they are combined with other surface ships, submarines and aircraft into Battle Groups (BG) under control of Composite Warfare Commanders (CWC). These BG Commanders need readiness status data and information to accomplish their required functions. Furthermore, BGs are combined into Battle Forces (BF) and BF Commanders must be provided with readiness data and information to support their decision making requirements. Finally, National Command Authority (NCA) must be kept apprised of the readiness status of all units.

If the total ship (comprised of a combat system and a hull, mechanical and electrical (HM&E) system) does not have adequate readiness information available at its interface with the BG, the BG Commander cannot be provided with the required readiness status, i.e. "a chain is only as strong as its weakest link."

The Shipboard Readiness Reporting System (SRRS) involves readiness assessment and reporting at the total ship level, improved by integrating readiness reporting and assessment of the combat and hull, mechanical and electrical systems comprising the total ship. The two areas of concentration in SRRS are "levels of reporting" and "data distribution." The area of "Levels of reporting" is emphasized in this paper.

2. INTRODUCTION

The Shipboard Readiness Reporting System (SRRS) will improve readiness reporting and assessment in surface combatants (missile launching capable surface ships). The SRRS is applicable to both new construction and in-service surface combatants.

Surface combatants are more complex now than ever before because:

(a) the threats to these ships have increased in quantity, capabilities and sophistication and the ships must be capable of coping with the increased threat,

(b) the surface combatant is combined with other ships, submarines, aircraft and land and space assets to form coordinated/cooperative Battle Groups and Battle Forces and the ships' design must accommodate these combined coordinated/cooperative operations and

(c) ship's spaces, systems and personnel are widely distributed throughout the ship for survivability and other reasons which creates new operational and maintenance problems and magnifies existing problems.

In Navy surface ships there are several tactical (operational) and technical (maintenance) spaces separated by relatively large distances. Examples include; (a) Combat Information Center (CIC), (b) a central location for controlling maintenance, (c) Damage Control Central (DCC) and (d) Work Centers (where operational equipment such as radars and sonars are located). Operational and maintenance readiness status data must be shared among these spaces in real (or near real) time using a common data base. These spaces could be linked via one or more local area networks (LAN) thereby facilitating the distribution of mission-specific doctrine, configuration alternatives, test schedules and scenarios and maintenance, mode, state and configuration reports. These data should be appropriately formatted, stored in a common data base, filtered in accordance with users' needs and then provided to tactical and technical users in a timely fashion.

Three technical problems are addressed by the SRRS:

(a) Accurate assessment of surface combatant capability and required corrective maintenance is difficult and time-consuming.

(b) There are no design standards for testing and subsequently reporting readiness status and there is no consistent methodology for collecting, formatting, distributing and displaying readiness status reports.

(c) As spaces, equipment and personnel are separated throughout the ship to improve survivability, data distribution (communications) must be improved to sustain proper operations and maintenance.

3. READINESS REPORTING PATHS

A U. S. Navy surface combatant is part of a readiness reporting and assessment hierarchical structure (architecture) that extends in both an upward and downward direction from the ship. Figure 1 shows this tiered hierarchical structure.

The reporting path above the ship includes the Battle Group (BG), Battle Force (BF) and National Command Authority (NCA). Below the ship are the systems, elements, equipments, cabinets, chassis, printed circuit boards (modules) and the components mounted on those modules. The ordering above and below the ship is important and the items within each level must be correctly identified. This is accomplished as part of the levels of reporting portion

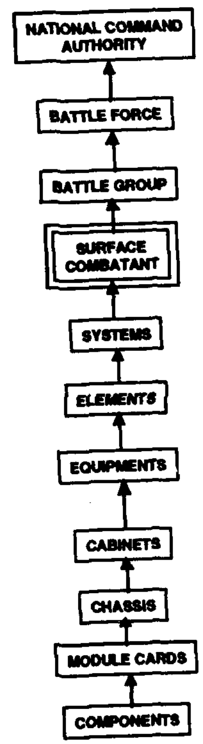


FIGURE 1. READINESS REPORTING AND ASSESSMENT HIERARCHICAL STRUCTURE

of SRRS.

It is a premise in this paper that any readiness status reporting above the ship, for use by higher authority, can be no better than what is available from within the ship, i.e., available at the ship/BG interface. The reason for this premise is that "a chain is only as strong as its weakest link." If a ship doesn't have its own complete readiness status, it can't very well report it to higher authority. There are programs that are attempting to improve readiness reporting and assessment above the ship level using artificial intelligence, data fusion and other techniques. However, if there are readiness reporting and assessment shortcomings within the ship, they must be corrected at the source of the problem (within the ship) to ensure that complete, correct, accurate and timely readiness reports can be made to higher authority.

4. CURRENT SRRS IMPLEMENTATION ANALYSES

Eventually, SRRS could be integrated into all surface ships, not just surface combatants, and could even be extended to cover other types of platforms and units. Figure 2 provides an overall Navy readiness reporting and assessment picture from National Command Authority down to the material, personnel and logistics readiness of each ship's constituent system.

Currently, SRRS is being applied to the areas shown vertically along the left side of Figure 2. SRRS is initially being applied to the material readiness of the combat system in surface combatants. Selected threads in specific Naval Warfare Mission Areas have been completed as part of the levels of reporting portion of SRRS. As the combat system thread analysis is completed, the results can be combined with similar hull, mechanical and electrical (HM&E) efforts ongoing at the David Taylor Research Center (DTRC). The combination of the combat and HM&E systems will complete the total ship, since these are the two constituent systems comprising a surface combatant.

5. SRRS CONSTITUENTS

The SRRS is being developed in two parts. One part is "levels of reporting" and the second part is "data distribution." Integration of these two parts of SRRS is being accomplished during all phases of SRRS development.

Levels of reporting involves identifying each level of the ship's systems hierarchical structure and the test requirements at each level needed to ensure appropriate readiness reporting to all system users (operators and maintainers).

Data distribution involves identifying all system nodes, the necessary fusion of readiness data and information at each system node and the data distribution (communications) interfaces between system nodes. The combination of message protocol definition, establishing interface requirements, identifying system nodes and precisely defining message traffic based on

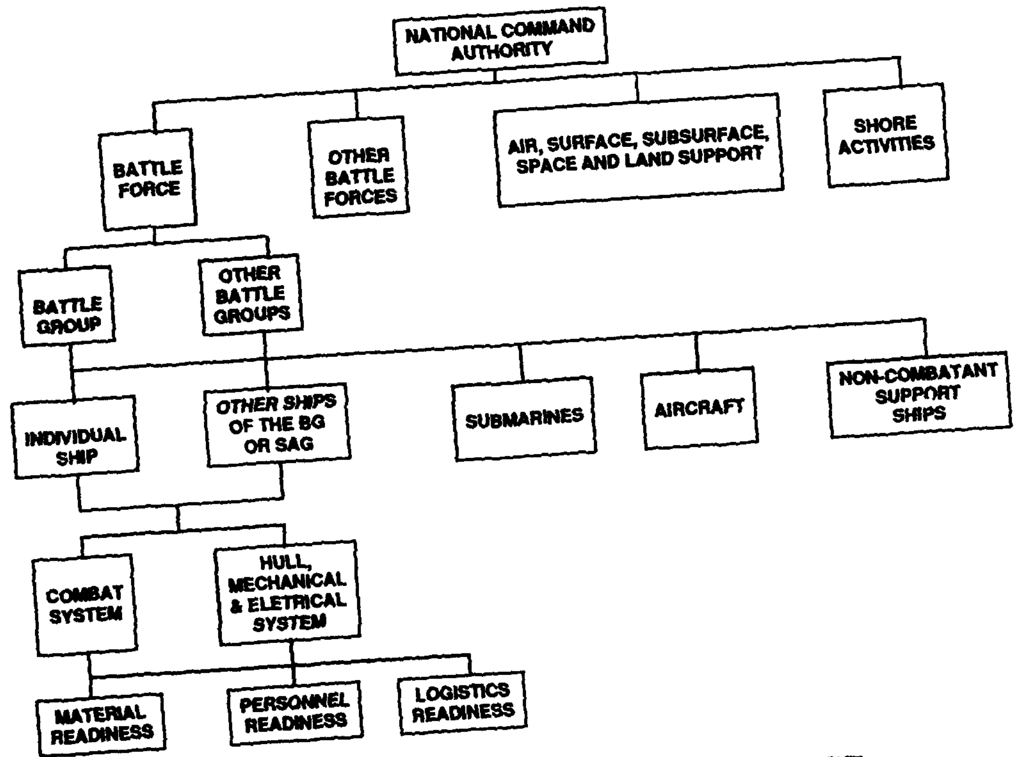


FIGURE 2. OVERALL NAVY READINESS REPORTING AND ASSESSMENT

users' needs should result in the right kind, and right amount, of readiness data and information being provided to each user in a timely fashion. This paper deals primarily with the levels of reporting portion of SRRS.

6. MISSION AREAS, CAPABILITIES, FUNCTIONS AND OBJECTIVES

Frequently, mission areas, capabilities, functions and objectives are mixed and merged with elements and equipments when a readiness reporting and assessment hierarchy (architecture) is being developed. These elements and equipments support the mission areas, provide the capabilities, perform the functions and accomplish the objectives. This mixing results in inappropriate positioning of many items in the surface combatant hierarchy. It is important to be able to distinguish between mission areas, capabilities, functions and objectives and Figure 3 attempts to sort this out.

Figure 3 shows the Naval Warfare Mission Areas across the top horizontal row. The middle horizontal row illustrates the common functions used in each warfare area and the bottom horizontal row contains the detailed objectives of each function. Figure 3 is not sufficiently detailed to completely distinguish between mission areas, capabilities, functions, objectives, elements and equipments. Therefore, Figure 4 was prepared to complete the picture by relating the functions and functional groupings to the elements and equipments that accomplish the functions.

7. COMBAT SYSTEM GENERAL HIERARCHICAL STRUCTURE

In order to provide proof of concept and reduce task accomplishment cost and manpower to manageable levels, the combat system was initially emphasized for SRRS. Figure 5 indicates the exponential growth in numbers of combat system constituents in a tiered hierarchical structure. While this hierarchical structure is complex, it reflects an actual combat system functional order and must be completed before proceeding further. The bad news is that each system must be precisely and correctly structured in a format that successively includes system, elements, equipments, cabinets, chassis, LRUs, modules and components. The good news is that this only has to be done once for each system and updated only to indicate system modifications.

Ship's readiness status data and information is derived from test results of the ship's systems at various levels within the system's hierarchical architecture (system, element, equipment, cabinet, chassis, etc.). Tests are conducted at all levels of the combat system from the system down to the components mounted on module cards and chassis. Test results are then used to report readiness status for both the operation and maintenance of the combat system. As part of SRRS, a typical combat system "top-down" architecture was developed so that the impact of faults and errors at low levels could be assessed at all higher levels (operational readiness status reporting). Also, when fault and error symptoms are indicated at any level of the combat system hierarchical structure, fault diagnosis is required at lower levels to isolate and correct the fault (maintenance readiness status reporting). Since it is

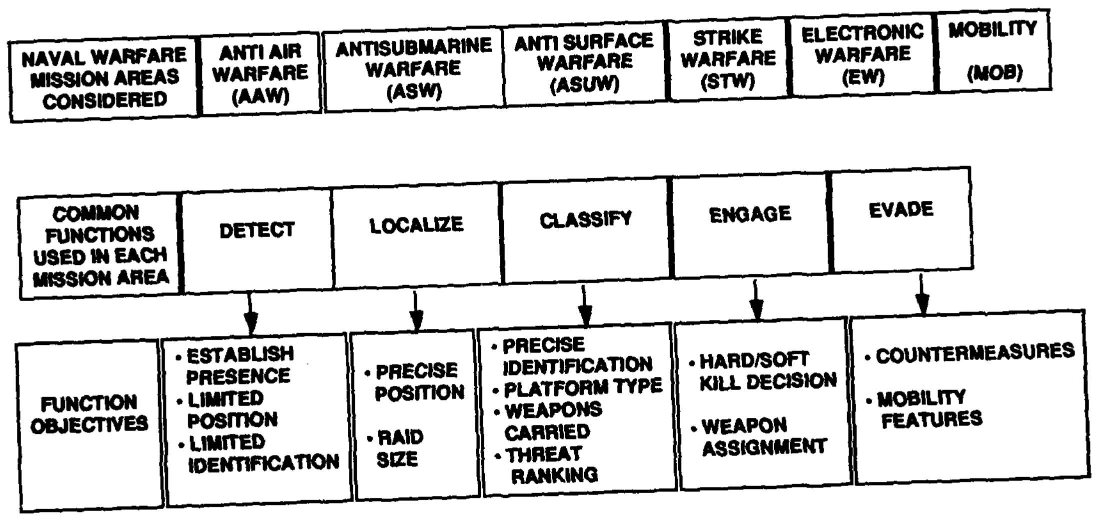


FIGURE 3. MISSION AREAS, FUNCTIONS AND OBJECTIVES

FUNCTIONS	FUNCTIONAL GROUPINGS	ELEMENTS	EQUIPMENTS
DETECTION	SENSORS	SEARCH RADARS	AIR SEARCH RADARS SURFACE SEARCH RADARS MULTIFUNCTION RADARS ELECTRONIC SUPPORT MEASURES EQUIPMENT
		SONARS	ACTIVE/PASSIVE SONARS ACTIVE SONARS PASSIVE SONARS
IDENTIFICATION	COMMAND & CONTROL	COMMAND & DECISION	ENGAGEMENT CONTROL EQUIPMENT
	COMMUNICA-TIONS	INTER/INTRA COMMUNICATIONS EQUIPMENT	DIRECT ENCRYPTED VOICE/DATA LINKS RELAY ENCRYPTED VOICE/DATA LINKS UNDERWATER COMMUNICATIONS EQUIPMENT INTERNAL COMMUNICATIONS EQUIPMENT AIRCRAFT LINK AND PROCESSING SHIPBOARD EQUIPMENT
ENGAGEMENT	WEAPONS	WEAPON POINTING ACCURACY EQUIP.	MULTIFUNCTION RADARS FIRE CONTROL RADARS
		ARMAMENT	LAUNCHERS GUN MOUNTS TORPEDO TUBES
		AMMUNITION	MISSILES TORPEDOES DEPTH CHARGES
EVASION	COUNTER-MEASURES	ACTIVE/PASSIVE COUNTERMEASURES	ACOUSTIC DECOYS CHAFF ACTIVE ELECTRONIC COUNTERMEASURES

FIGURE 4. FUNCTIONAL RELATIONSHIP TO ELEMENTS ANDEQUIPMENTS

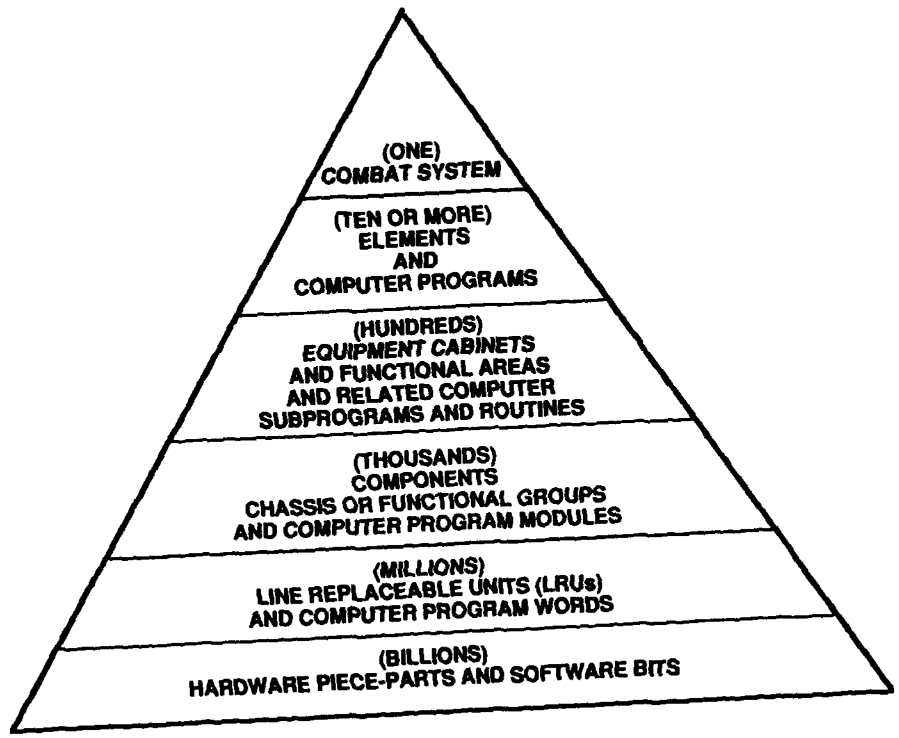


FIGURE 5. COMBAT SYSTEM HIERARCHY

necessary to be able to go both up and down from any level within the combat system hierarchy whenever a fault or error symptom occurs, it is essential that the combat system hierarchical structure be defined in advance of the design process. This will also allow identification of levels which require additional testing, levels in which redundant testing exists, and "hardspots" from a testing and readiness status reporting point of view.

8. COMBAT SYSTEM READINESS REPORTING AND ASSESSMENT PROBLEMS

Current combat systems in surface combatants have readiness reporting and assessment problems inherent in their design. SRRS must avoid these problems in future combat system designs and correct them in existing fleet ships. Examples include:

(a) Inability to differentiate between an equipment that has failed, an equipment that is in other than the normal operational mode or state and an equipment that is not fully initialized.

(b) Testing periodicity and fault detection and isolation coverage is substantially different from equipment to equipment and this is frequently not accounted for in current readiness reporting and assessment approaches.

(c) Operational and maintenance data and information are mixed and indiscriminately provided to both tactical (operational) and technical (maintenance) personnel. This is confusing and the data needs to be filtered to make it more meaningful to the user and more concisely presented.

(d) The readiness terminology used varies widely from equipment to equipment, so that different terms are used to mean precisely the same thing.

9. COMBAT SYSTEM THREAD

For purposes of this paper, the SRRS levels of reporting methodology is illustrated using a single thread within the combat system. The Antisubmarine Warfare (ASW) mission area was selected for the illustration. The selected thread extends from the combat system level down to the ASW Naval Warfare Mission Area. Then, an overall ASW hierarchy was developed to identify the ASW constituents at each level. Next, the thread extends down from ASW to the hull mounted and towed array sonars in which both active and passive detections are included. Finally, a detailed functional thread called "Localization of ASW Contacts Using Sonobuoys" is shown and described.

9.1 Combat system to ASW

Currently, the combat system is pretty well partitioned in terms of the Naval Warfare Mission Areas that it supports. There are procedures and doctrine that accurately localize problems to the specific Naval Warfare Mission area that contains the problem. Figure 6 is an overall ASW hierarchy identifying the ASW functions (detect, localize, etc.) across the top row. The next

DETECT	LOCALIZE	CLASSIFY	ATTACK	DESTROY	AVOID ENEMY TORPEDOES
SENSORS ELEMENT	DISPLAY SHARING/COMPUTER SWITCHING ELEMENT		WEAPONS ELEMENT		TORPEDO COUNTERMEASURES ELEMENT
ACTIVE SONAR EQUIPMENT	PASSIVE SONAR EQUIPMENT	SHIPBOARD SONOBUOY EQUIPMENT	CONTROL EQUIPMENT	TORPEDO LAUNCHING EQUIPMENT	AUDIO EQUIP.
				DEPTH CHARGE LAUNCHING EQUIP	BUBBLE MAKING EQUIP
AN/SQS-53C (ACTIVE)	AN/SQS-53C (PASSIVE) AN/SQR-19 (PASSIVE)	AN/SQQ-28 AN/SRQ-4 AN/SKR-4 AN/ARR-75 LAMPS MK I LAMPS MK III	AN/SQQ-89 MK 116 MOD 7 ICOM EXCOM	VLS/ASROC OTS TORPEDOES LAMPS MK I LAMPS MK III	AN/SLQ-25 (NIXIE) PRAIRIE MASKER

FIGURE 6. SURFACE COMBATANT ASW HIERARCHY

row identifies the ASW elements and the third row from the top generally identifies each equipment grouping. Finally, at the bottom of Figure 6, is a listing of each ASW equipment that belongs to each equipment grouping. The purpose of this ASW hierarchy is to ensure that nothing is omitted when developing the SRRS levels of reporting for the selected ASW thread.

9.2 ASW to sonars

The purpose of the sonars is to perform the ASW surveillance function. In other words, active and passive hull mounted and towed array sonars must detect underwater contacts. Figure 7 shows this portion of the selected combat system thread starting at the ASW Naval Warfare Mission Area and extending down to the hull mounted and towed array sonars.

Notice, in Figure 7, that there are other Naval Warfare Mission Areas, but this thread deals only with ASW. There are other functions, but this thread deals only with DETECT (limited localization and identification are accomplished as shown by dashed box). There are also other ASW detection capabilities, but this thread deals only with active and passive sonars.

9.3 Localization of ASW contacts using sonobuoys

Once an ASW contact has been either actively or passively detected using the hull mounted or towed array sonars, a decision might be made to localize the ASW contact using sonobuoys. Figure 8 contains this portion of the selected combat system thread and shows the required functions and equipments to localize ASW contacts using sonobuoys.

In order to Localize ASW Contacts Using Sonobuoys, it is necessary to

(a) DEPLOY SONOBUOYS,

(b) PROCESS SONOBUOY DATA FROM OWNSHIP, and

(c) PROCESS SONOBUOY DATA FROM OTHER SOURCES.

These functions are shown within dotted boxes in Figure 8. The equipments that accomplish the functions are shown within solid boxes. The equipments are laid out in a tiered hierarchical architecture dictated by how the outputs, inputs and interfaces are positioned during normal system operation. Functional dependency of one equipment on another is a prime consideration of the layout.

There are five sources for deploying sonobuoys; lamps MK I, lamps MK III, carrier based helicopters, fixed wing aircraft and ownship. The fixed wing aircraft include P-3s and S-3s. In order to deploy the sonobuoys effectively, the tactical and technical users must be advised, via readiness reporting, of the availability of each of the five sources for deploying sonobuoys.

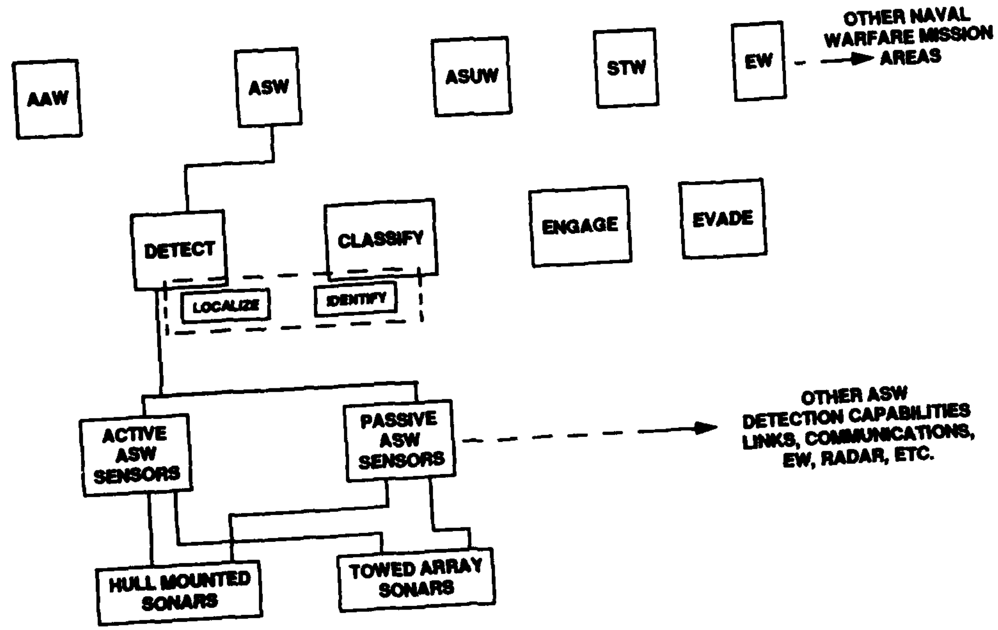


FIGURE 7. COMBAT SYSTEM THREAD - ASW TO SONARS

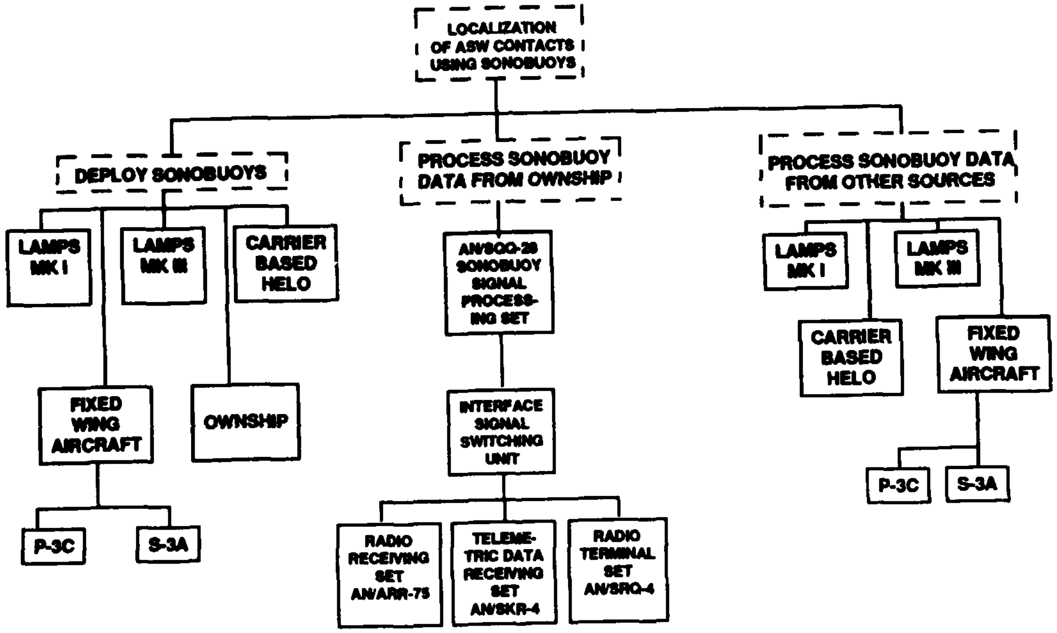


FIGURE 8. LOCALIZATION OF ASW CONTACTS USING SONOBUOYS

Once sonobuoys are deployed by one of the five sources, the ship must be capable of processing the data from the deployed sonobuoys. The equipments shown in Figure 8 (AN/SQQ-28, Interface Signal Switching Unit, AN/ARR-75, AN/SKR-4, AN/SRQ-4) all can play a role in processing sonobuoy data. The ARR-75 links the sonobuoys directly to the ship. The AN/SKR-4 is the link between lamps MK I and the ship while the AN/SRQ-4 is the link between lamps MK III and the ship. Once again, the tactical users must be informed of the operational readiness status of each of these three ASW equipments so that they can make intelligent decisions on how to best get sonobuoy data to the ship for processing. Notice that if the Interface Signal Switching Unit or the AN/SQQ-28 is hard down, shipboard processing of sonobuoy data will not be possible.

There is a path, shown on the top right of Figure 8, that enables processing of sonobuoy data from other sources off-ship. This includes lamps MK I, lamps MK III, carrier based helicopters, and fixed wing aircraft (P-3s or S-3s).

The essence of this example is that the readiness status of each equipment to support each element must be known. Furthermore, the impact of any element or equipment problems must be assessed to determine current ship capabilities. Included in the capabilities category are deploy sonobuoys, process sonobuoy data, etc. as shown by the dashed boxes of Figure 8 while the solid boxes of Figure 8 represent elements and equipments.

10. TOOL FOR SIMPLIFYING THE HIERARCHY

Existing combat systems consist of hundreds of equipments. Each equipment falls into one of three categories.

Category 1. Test results are collected at the equipment level and are transmitted to the element (and higher levels);

Category 2. Test results are collected at the equipment level but are not transmitted to the element;

Category 3. Test results are not adequately collected at the equipment level.

The readiness data flow diagram, Figure 9, illustrates a method for analyzing and correcting (when required) existing systems and providing concise, accurate, and timely readiness reports to tactical and technical users. Test results from equipments in Category 1 only require minor formatting before entry into a common data base and, as shown in Figure 9, the test results are sent directly to a DATA FORMATTER. Test results from Category 2 equipment require interface modifications between the equipment and higher levels. This is accomplished by the CREATE INTERFACE block in Figure 9. Equipment in Category 3 must be modified to collect the test results and the interface must also be modified to pass these test results to the DATA FORMATTER. Note that the SRRS, a product of top-down design, would result in Cate-

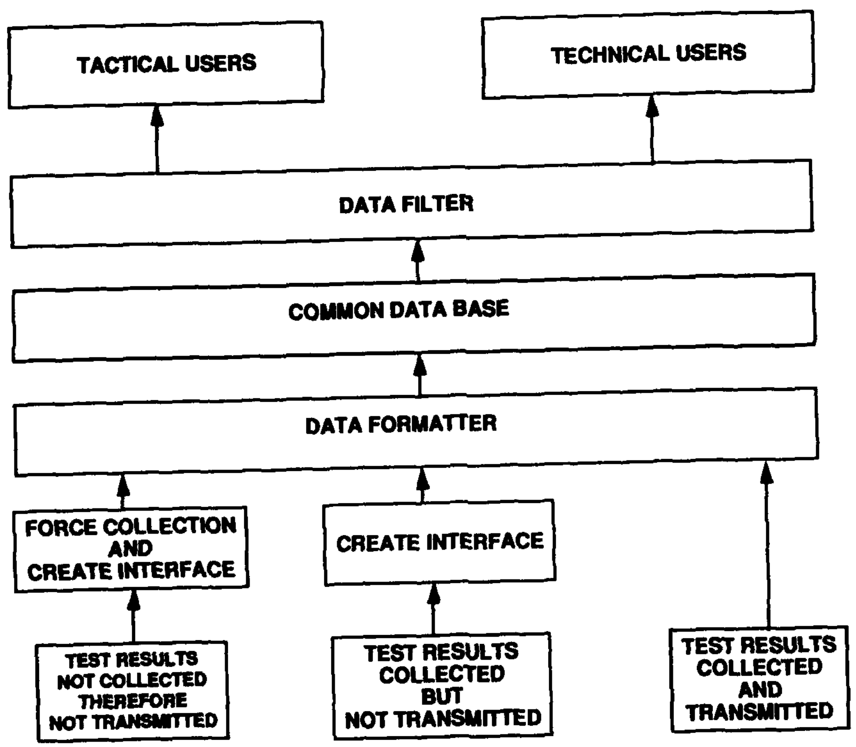


FIGURE 9. READINESS DATA FLOW DIAGRAM

gory 1 equipments for newly designed combat systems and would identify Category 2 and 3 equipments for modification for in-service combat systems.

The DATA FORMATTER corrects and standardizes inputs to the COMMON DATA BASE. The DATA FILTER then organizes the readiness related data and informatmion into two groups; one that satisfies tactical (operational) user requirements and a second that satisfies technical (maintenance) user requirements. Finally, appropriate readiness data and information are delivered for display to tactical and technical users in a concise, accurate and timely fashion.

11. FUTURE PLANS

The intent is to expand SRRS both horizontally and vertically and to transition SRRS from its current 6.2 (Exploratory Development) status. The horizontal expansion entails completing other ASW threads, completing combat system threads in other Naval Warfare Mission Areas and, finally, integrating combat system threads with similar threads developed for the HM&E system. The vertical expansion of SRRS involves integrating the SRRS levels of reporting effort with ongoing fault detection, diagnostics and isolation efforts. This should lead to a complete top to bottom hierarchical architecture and best use of test results at all levels for operational and maintenance readiness status reporting.

11.1 Horizontal expansion

Most of the horizontal expansion of SRRS involves completing other ASW threads and developing combat system threads in other Naval Warfare Mission Areas (AAW, ASUW, STW, EW, etc.). However, the most important horizontal expansion of SRRS involves the integration of readiness reporting and assessment of the combat and hull, mechanical and electrical (HM&E) systems. NOSC is accomplishing the SRRS task under the auspices of the Office of Naval Technology (ONT) Code 226. The David Taylor Research Center (DTRC) is working on a task called Condition Based Maintenance (CBM) for HM&E equipment under the same 6.2 block funding that covers SRRS. Both NOSC and DTRC are involved with the NAVSEA sponsored Damage Control Management System (DCMS) that might serve as a vehicle to transition SRRS. This could result in the combat and HM&E systems being integrated from a readiness reporting and assessment point of view.

11.2 Vertical expansion

Some of the threads developed as part of SRRS levels of reporting are not yet complete through all levels of the hierarchy. The Naval Research Laboratory (NRL) is working on a 6.2 block funded task involving artificial intelligence applications of fault isolation diagnostics for the AN/SQS-53 Hull Mounted Sonar. This could serve as an excellent vertical expansion of a specific ASW thread if these two portions of an overall thread could be properly integrated.

Other related programs include: the AEGIS Operational Readiness Test System (ORTS), the Combat System Technical Operations Manual (CSTOM), the Engineering and Combat System Operational Sequencing Systems (EOSS and CSOSS - separately and independently developed), the Integrated Diagnostics Support System (IDSS) and many other programs too numerous to mention here. These would all support vertical expansion of SRRS.

11.3 Transition

It is a goal of every program and project at some point in time, to transition toward eventual in-service use and SRRS is no exception. The Damage Control Management System (DCMS), directed by NAVSEA, seems to be an ideal program to transition SRRS into. DCMS is principally used in combat and especially after a ship has sustained battle damage. Under these conditions there is no time for repairing failures through long corrective procedures like removal/replacement or alignment. DCMS must continuously know the precise readiness status of surface combatant systems and also must know which rapid corrective action can be taken or what alternate resources can be employed.

SRRS covers the entire readiness reporting and assessment picture including areas of fault tolerance and rapid recovery. DCMS only has time for rapid recovery corrective action where fault tolerant design features are provided. These functions must be identified within the SRRS levels of reporting and integrated into DCMS. Figure 10 indicates a sorting method that could be used for this purpose, thereby taking the first steps of integrating SRRS with DCMS.

HARDWARE FAULTS		SOFTWARE ERRORS	
RECOVERABLE	**NON-RECOVERABLE**	**RECOVERABLE**	**NON-RECOVERABLE**
RECONFIGURATION	CRITICAL FAILURE	RELOAD (SAME PROGRAM)	REMOTE PATCH
REDUNDANCY	MAJOR FAILURE	RELOAD (BACK UP PROGRAM)	PROGRAM UPDATE
CHANNELIZATION	MINOR FAILURE	LOCAL PATCH	
LOAD SHARING			
RECYCLING			

FIGURE 10. SYSTEM MALFUNCTIONS SORTED BY RECOVERY TIMES

IDENTIFICATION AND COMPUTER CONTROL OF A TURBOCHARGED MARINE DIESEL ENGINE

by John D. Forrest
Royal Naval Engineering College
Manadon, Plymouth, UK

1. ABSTRACT

It is well known that the dynamic characteristics of turbocharged diesel engines can vary greatly over their load/speed operating map. The current generation of Royal Navy prime movers are fitted with simple hydraulic speed governors which take little account of these varying characteristics. This paper describes the identification of the dynamic response of a 6 cylinder turbocharged marine diesel engine using parameter estimation techniques. Experimental response data obtained from a number of PRBS tests on the engine is analysed using both the generalised least square and instrumental variable algorithms. Continuous and discrete-time mathematical models of the engine relating fuel rack position to engine speed are identified for three different operating points and a comparison of the models produced using the two algorithms is given. The validity of the models is investigated and a comparison is made between the model predicted and test-bed responses of the engine. The implementation of a simple 3-term microcomputer-based controller is described and some typical test results are discussed. Finally, suggestions are made where the work might be extended to improve the control laws and utilise more fully the processing power available with a digital controller.

2. INTRODUCTION

The turbocharged diesel engine has been familiar to marine engineers for many years and it is well known that their dynamic characteristics can vary greatly with load and speed. (1,2) When used in generating applications, such engines are, in general, open-loop unstable and so require governing. The current generation of Royal Navy prime movers are fitted with simple hydraulic speed governors which take little account of these varying characteristics. Direct digital speed regulation

offers the opportunity of employing sophisticated controller algorithms which would be able to compensate for these variations and so produce improvements in both fuel economy and transient response.

The present study describes the identification of an instrumented engine at various points within the load/speed operating range by applying parameter estimation techniques to the response data obtained from PRBS tests on the engine. The family of linear models produced by this technique can then be used to design a suitable digital controller for the engine which can be implemented via a microcomputer.

The paper begins with a description of the engine test bed. This is followed by Section 4 which describes the engine identification procedure and contains the identification test results obtained at various engine loads and speeds. Section 5 describes the design, implementation and testing on the engine of a simple three term digital controller. Section 6 concludes the paper by suggesting areas where the work reported may be further developed.

3. THE TEST BED

The engine used in this study was a six cylinder, two stroke, medium speed, turbocharged FODEN FD7 Diesel installed in a test cell at the Royal Naval Engineering College, Manadon, Plymouth. The engine is equipped with a Holset Mark 4 turbocharger of the single-stage centrifugal compressor/radial turbine type.

The engine operating speed is in the nominal range of 800 rpm to 2200 rpm and a typical loading is 700 Newton metres. The engine is equipped with a hydraulic speed governor, a Froude water brake and the appropriate instrumentation to monitor the signals required for the identification study. These were:

- demanded speed
- actual engine speed
- fuel rack position
- engine load

The actual engine speed was measured using a toothed wheel mounted on the end of the transmission shaft and an active magnetic proximity transducer giving a once per tooth pulse signal. The resulting pulse train frequency was converted to a proportional analogue voltage using a tachometer integrated circuit. The output signal from the converter was subject to ripple at the frequency of the shaft rotation and this was removed by inserting a 100pF capacitor across the signal line.

The position of the fuel rack was measured by a linear potentiometer mounted immediately above the rack and operated by an arm directly attached to the rack. This transducer gave good undistorted measurements of the rack displacement and was used directly.

The engine/load torque was measured using a load cell together with related signal conditioning equipment.

Signals were injected into the engine via a pivoted lever arm connected at one end to the engine fuel rack. The other end of the lever arm was connected to a 240V electrical solenoid. This arrangement allowed both step and pseudo random binary sequence (PRBS) perturbations to be applied directly to the fuel rack.

The test bed computer was a Kemitron (K2000ED) general purpose laboratory microcomputer which was equipped with in-built analogue to digital and digital to analogue convertors. The data acquisition software was written in PASCAL and the sampled data was recorded on a floppy disc for later transfer to a CYBER (180/840) mainframe computer.

4. IDENTIFICATION

4.1 Theoretical background

A wide range of techniques are available for determining the dynamic characteristics of a diesel engine (3-9) but for this study it was decided to base the identification phase upon parameter estimation techniques.

These techniques assume that the system to be identified can be represented by a model of the form shown in Figure 1.

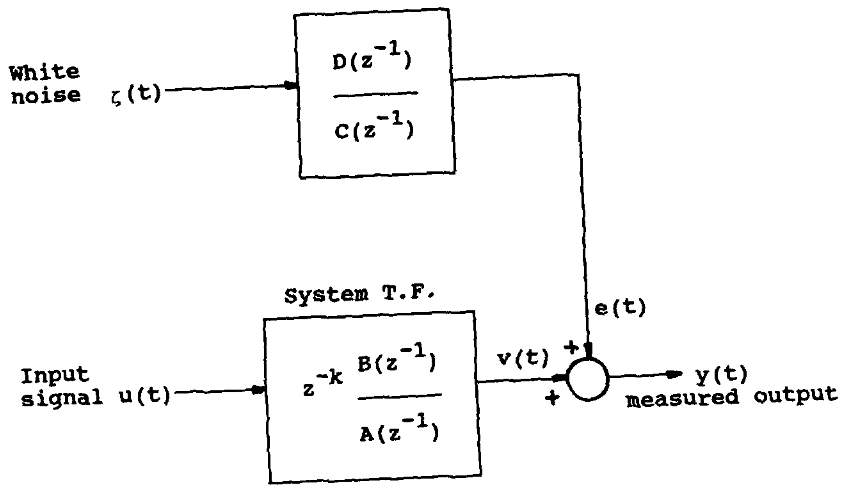


Figure 1 Representation of the system

The transfer function models obtained are of a discrete-time, z-transform structure:

$$y(t) = z^{-k} B/A\ u(t) + C/D\zeta(t), \text{ or:} \tag{1}$$

$$y(t) = z^{-k} \frac{B(z^{-1})}{A(z^{-1})} u(t) + \frac{D(z^{-1})}{C(z^{-1})} \zeta(t) \tag{2}$$

where: z is the forward shift operator
$u(t)$ is the system input at sample time t
$y(t)$ is the system output at sample time t
$\zeta(t)$ is a 'white noise' signal

The polynomials A, B, C, D are given by:

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots\dots\dots + a_n z^{-n} \tag{3}$$

$$B(z^{-1}) = \quad b_1 z^{-1} + \dots\dots\dots + b_n z^{-n} \tag{4}$$

$$C(z^{-1}) = 1 + c_1 z^{-1} + \dots\dots\dots + c_m z^{-m} \tag{5}$$

$$D(z^{-1}) = 1 + d_1 z^{-1} + \dots\dots\dots + d_m z^{-m} \tag{6}$$

The model order is n, the noise-model order is m, and the time-delay is k. The term z^{-k} B/A is the system transfer function $G(z^{-1})$, and the term D/C is the noise transfer function $N(z^{-1})$ which is included to account for disturbances, measurement noise and model inaccuracies.

Rewriting equation (2) in terms of data sequences, the model is:

$$v(t) + a_1 v(t-1)+\ldots+a_n v(t-n) = b_1 u(t-k-1)+\ldots+b_n u(t-k-n) \tag{7}$$

$$e(t) + c_1 e(t-1)+\ldots+c_m e(t-m) = \zeta(t) + d_1\zeta(t-1)+\ldots+d_m\zeta(t-m) \tag{8}$$

$$y(t) = v(t) + e(t) \tag{9}$$

If now input/output data records $\{u(t),y(t);t=1\ldots N\}$ are obtained from an experiment on the plant then a suitable model may subsequently be estimated using an appropriate identification software package.

4.2 Experimental procedure

A natural starting point when considering the identification of an unknown system is to examine its step response. The step change is the most severe disturbance possible for any given signal amplitude and is a type of change which frequently occurs in practice. Step inputs are excellent for estimating the gain of a system and indicating the magnitude of the system time constants and also provide a useful guide to the selection of suitable parameters for the PRBS test signal.

Following the completion of such step tests the test signal chosen for the identification study was a PRBS signal of length 127 bits and bit interval 1.0 second. The test perturbation signal was applied directly to the fuel rack with the engine governor overridden. Tests were conducted on the engine at low, medium and high speeds which also corresponded to low, medium and high engine loads as shown in Table 1.

Table 1   Selected operating points for identification study

	Speed (rpm)	Load (Nm)
Low	1200	447
Medium	1600	668
High	2000	887

During each identification test the Kemitron microcomputer recorded 500 samples of both the PRBS input signal and the engine speed response at a sampling interval of 0.1 seconds.

The recorded data was transferred to the College CYBER 840 mainframe computer for subsequent analysis by a system identification software package acquired from the University of Oxford (10). This package offers the user a variety of identification algorithms including generalised least square, instrumental-variables and maximum likelihood. The package also contains diagnostic routines which indicate the validity of the estimated models together with a comprehensive graphical data-display program.

4.3 Test results

The various engine response data files were analysed using both the generalised least squares algorithm and the instrumental-variable algorithm. This analysis involved estimating the coefficients in linear models of varying order (n=1, 2, 3) and with various time delays (k=0, 1, 2) and various noise model orders (m = 0, 1, 2, 3).

In order to determine the most appropriate model to be fitted to a particular data file, the following questions were asked of each model. In each case the answers were provided by the diagnostic routines within the package.

1. Is the mean-square error of the current model significantly less than that of the previous lower order model?

   The answer to this question was provided by the F-ratio test. This statistic compares the variance of the residuals with that of the next lower order model and determines whether it is significant (ie whether the new model gives a 'better' fit to the data).

2. Are the standard deviations of the parameters small in comparison to the estimated values?

   Unfortunately, the IV algorithm does not produce standard deviations and so this test could not be used there.

3. Are the residuals white?

   If the model is correct the residuals should be entirely white-noise. The diagnostic software computed their autocorrelation function (ACF), which should be

impulse like and counted points outside the 95% confidence limits.

4. Are the residuals independent of the input?

   As the residuals should depend upon the disturbances only, there should be no dependence here if the model order and time delay are correct. The diagnostic software computed the cross-correlation function (CCF) between the input and the residuals and once again counted points outside the 95% confidence limits.

5. Does visual inspection of the predicted output reveal significant discrepancies when compared with the actual (experimental) output?

6. Does the system gain as estimated by the step tests on the engine compare favourably with the gain of the estimated models?

A consideration of the above questions indicated that for both the low speed, low load and the medium speed, medium load operating points, second order models were adequate in explaining the data. In the case of the high speed, high load operating point, first order models were found to be adequate. Furthermore, in all cases a first order noise model was also found to be adequate.

The models obtained for each operating point using the two different parameter estimation algorithms are shown in Table 2 and Table 3. For each case the table shows three models comprising:

1. The z-domain engine transfer function plus noise transfer function derived from the parameter estimates and in the form of Equation (2).

2. The z-domain transfer function of the engine dynamics alone, $G(z)$.

3. The equivalent continuous transfer function of the engine dynamics, $G(s)$.

Table 2 Generalised least squares algorithm

Low speed:

$$y(t) = \frac{0.0334z^{-1}+0.0096z^{-2}}{1-1.188z^{-1}+0.2703z^{-2}} u(t) + \frac{1-0.2415z^{-2}}{1-0.8178z^{-2}} \zeta(t)$$

$$G(z) = \frac{0.0334(z+0.288)}{(z-0.8810)(z-0.3068)}$$

$$G(s) = \frac{0.5215(1+0.017s)}{(1+0.789s)(1+0.084s)}$$

Medium speed:

$$y(t) = \frac{0.0444z^{-1}+0.0188z^{-2}}{1-1.012z^{-1}+0.0931z^{-2}} u(t) + \frac{1-0.4529z^{-1}}{1-0.8475z^{-1}} \zeta(t)$$

$$G(z) = \frac{0.0444(z+0.4237)}{(z-0.9091)(z-0.1024)}$$

$$G(s) = \frac{0.7756(1+0.002s)}{(1+1.049s)(1+0.044s)}$$

High speed:

$$y(t) = \frac{0.0997z^{-1}}{1-0.8854z^{-1}} u(t) + \frac{1-0.3608z^{-1}}{1-0.7946z^{-1}} \zeta(t)$$

$$G(z) = \frac{0.0997}{z-0.8854}$$

$$G(s) = \frac{0.8705}{1-0.8216s}$$

Table 3 Instrumental - variable algorithm

Low speed:

$$y(t) = \frac{0.0336z^{-1}+0.0281z^{-2}}{1-0.770z^{-1}-0.1165z^{-2}}\,u(t) + \frac{1-0.2560z^{-1}}{1-0.8097z^{-1}}\,\zeta(t)$$

$$G(z) = \frac{0.0336(z+0.836)}{(z-0.8996)(z+0.1295)}$$

$$G(s) = \frac{0.5444(1+0.0153s)}{(1+0.945s)(1+0.49s)}$$

Medium speed:

$$y(t) = \frac{0.0456z^{-1}+0.0492z^{-2}}{1-0.5527z^{-1}-0.3219z^{-2}}\,u(t) + \frac{1-0.4532z^{-1}}{1-0.8477z^{-1}}\,\zeta(t)$$

$$G(z) = \frac{0.0456(z+1.079)}{(z-0.9074)(z+0.3547)}$$

$$G(s) = \frac{0.7561(1+0.073s)}{(1+1.029s)(1+0.096s)}$$

High speed:

$$y(t) = \frac{0.09552z^{-1}}{1-0.8875z^{-1}}\,u(t) + \frac{1-0.3644z^{-1}}{1-0.8016z^{-1}}\,\zeta(t)$$

$$G(z) = \frac{0.0955}{z-0.8875}$$

$$G(s) = \frac{0.8489}{1+0.838s}$$

A comparison of the engine response output and predicted response output to the PRBS input signal is shown in Figure 2 for the medium speed condition using the generalised least squares algorithm. As can be seen, correlation between the actual and predicted responses is good and similar results were obtained for the two other models.

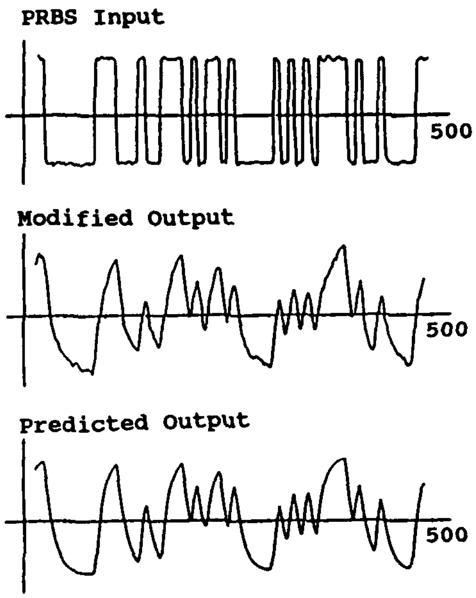


Figure 2 PRBS input, speed output and predicted output for medium speed diesel engine over 500 data points

5. DIESEL ENGINE CONTROL

5.1 General

The use of the electronic speed control governor has been growing rapidly in recent years as designers have begun to recognise their potential advantages (11). The theoretical advantages of digital control are detailed in most texts on the subject (12) but a digital implementation of the governor has some specific advantages for the improvement of diesel engine performance. These include the reduction of exhaust emissions

and the improvement of fuel economy brought about by running the engine as close as possible to the optimum operating point, together with an improved transient response to both speed and load changes. Moreover, the controller is easily incorporated into a wider based engine management system concerned with engine health monitoring and surveillance.

5.2 Controller design

A relatively simple approach to the design of a suitable digital controller was adopted for this study, based around a discrete version of the classical three term or PID algorithm. These algorithms are deservedly popular in industrial control systems, they give satisfactory performance for a wide class of processes and they are easy to implement using analogue or digital hardware.

The classical PID algorithm has the following form:

$$m(t) = K_c\left[e(t) + \frac{1}{T_i}\int e(t)dt + T_d\,\frac{de(t)}{dt}\right] \qquad (10)$$

where $m(t)$ = control signal

$e(t)$ = error signal

K_c = controller gain

T_i = integral action time

T_d = derivative action time

Equation (10) can be transformed into the Laplace domain to give the transfer function representation:

$$G_c(s) = \frac{M(s)}{E(s)} = K_c\left[1 + \frac{1}{T_i s} + T_d s\right] \qquad (11)$$

Equation (11) implies that in the steady-state, with zero error, the controlled variable M is zero. For many applications and in particular for the diesel engine, this is not true. Under these circumstances it is normally the practice to modify equation (11) by the addition of a constant term, MR, representing the value of the manipulated variable in the steady-state, thus giving

$$G_c(s) = \frac{M(s)}{E(s)} = K_c\left[1 + \frac{1}{T_i s} + T_d s\right] + MR \tag{12}$$

The quantity MR can be thought of as setting the operating point for the controller. If it is omitted, then, if integral action is present, the integral action term will compensate for its omission, but there will be difficulties in changing smoothly, without disturbance to the plant, from manual to automatic control.

For computer implementation, equation (12) must be written in discrete form. A straightforward discretisation of Equation (12) using the backward difference approximation leads to:

$$m_k = K_c\left[e_k + \frac{T.s_k}{T_i} + \frac{T_d}{T}(e_k - e_{k-1})\right] + MR \tag{13}$$

with $s_k = s_{k-1} + e_k$ being the sum of errors (14)

By introducing new parameters as follows:

$$K_p = K_c \tag{15}$$

$$K_i = \frac{K_c T}{T_i} \tag{16}$$

$$K_d = \frac{K_c T_d}{T} \tag{17}$$

equation (13) can be expressed as an algorithm of the form:

$$m_k = K_p e(k) + K_i s(k) + K_d[e(k) - e(k-1)] + MR \tag{18}$$

The digital controller was developed remotely from the engine on the Kemitron microcomputer interfaced to an analogue model of the diesel engine obtained from the earlier identification study. The particular model chosen was that derived for the medium speed, medium load operating point via the generalised least squares algorithm. The controller software was written in PASCAL and operated in real time on the Kemitron via its A/D and D/A convertors. Positioning of the engine fuel rack was achieved via a 12v dc precision motor/gearbox combination powered by a positional servo-control module. The precision motor actuated the fuel rack by means of a mechanical lever arm

arrangement. In order that the actuator dynamics could be included in the simulation loop the servo-actuator was correctly mounted on the engine and linked to the fuel rack and the analogue computer was installed in the test cell close to the engine. Using this arrangement a voltage signal proportional to fuel rack position could be taken from the fuel rack potentiometer. This signal formed the input to the analogue computer. The desired speed signal was derived from a variable voltage source. A block diagram of the arrangement is shown in Figure 3.

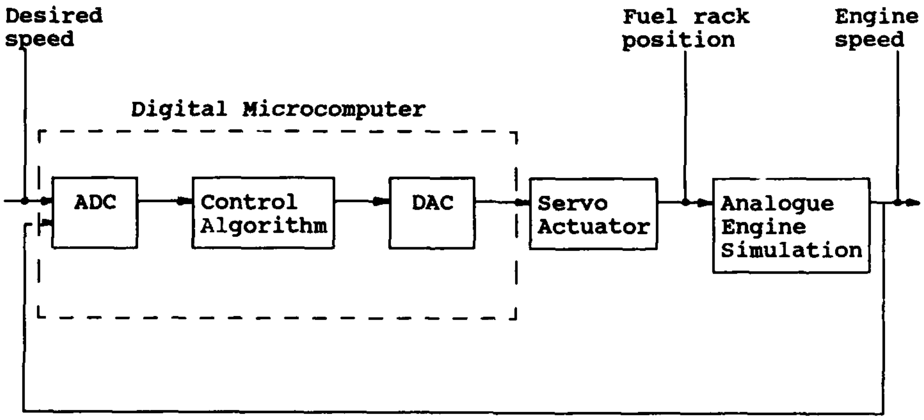


Figure 3 Block diagram of arrangement used during simulation testing

By means of this arrangement a suitable value for MR was established and the controller coefficients K_p, K_i and K_d were tuned using the Ziegler-Nichols ultimate sensitivity method (13). This method is valid for use with discrete controllers provided the sample interval is short and various rules of thumb have been proposed for choosing a sample interval while maintaining the analogue controller parameters (14, 15). Following satisfactory testing on the analogue simulation, the controller was installed on the Diesel engine.

5.3 Test Results

The preliminary analogue simulation tests proved their worth when the digital controller was transferred to the diesel engine

and quickly set to work. Experimental tests on the performance of the digital controller were confined to its operation about the medium speed, medium load operating point. A block diagram showing the arrangement used to test the controller on the engine is shown in Figure 4.

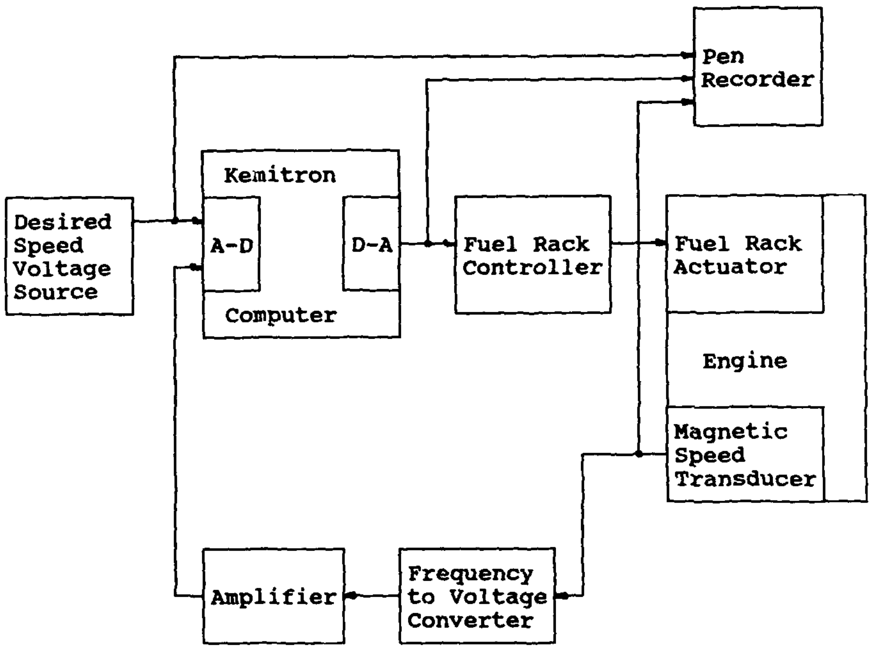


Figure 4 Block diagram showing system hardware and interfaces used to test controller on the engine.

The Ziegler-Nichols ultimate sensitivity test was repeated at this stage, so enabling the controller parameters derived from the analogue simulation to be further improved. The testing of the controller produced a large number of results which are fully discussed elsewhere (16) and only a sample are considered in the following results.

a. An Example of P + I Control An example of a typical step response produced by a two term PI controller with the

engine operating about the medium speed, medium load operating point is shown in Figure 5. A step change in engine speed of approximately 150 rpm was demanded by injecting a voltage step of 0.6V into the desired speed ADC channel of the Kemitron microcomputer. The controller coefficients were chosen to be K_p=4.5, K_i=0.421 and K_d=0 with a sampling interval of T = 0.07s.

Figure 5 Step response of Diesel engine using digital PI controller

The controller produced an acceptable transient performance in response to a step change in desired speed which can be seen to be direction dependent due to the engine dynamics and in particular the effects of turbocharger lag. The integral term in the algorithm successfully eliminated any steady-state error between demanded speed and actual speed. The addition of derivative action to the control algorithm greatly increased the control signal activity but did little to improve the transient performance.

<u>**b. Effect of Sampling Rate on Engine Response**</u> A proper selection of the sampling frequency for a digital control system is very important. The cost involved will be less if the sampling frequency is less, because with a slow sampling frequency the computations involved can be handled by a smaller and less expensive computer. However, although a slower sampling

frequency is less expensive, it generally degrades the system performance.

In practical situations, one approach to the selection of an appropriate sampling frequency is to choose it on the basis of the systems transient response rise time. A rule of thumb is to sample 10-20 times during the 95% rise time in response to a step input (14). From previous step response tests on the engine the 95% rise time had values of 1.2s and 1.7s depending upon the step direction. Applying the rule of thumb to these results suggest that a suitable value for the sampling interval T should lie within the range $0.06 \le T \le 0.17$s.

An alternative approach (15) suggests that a suitable sample interval for the system should be approximately $T=0.1T_p$ where T_p is the period time measured during the Ziegler-Nichols ultimate-sensitivity method. The value of T_p obtained during the Ziegler-Nichols tests on the engine was 0.9s. This leads to an approximate value for T of 0.09s, which is within the range suggested by the first approach.

A series of step tests were carried out in order to examine the effect on the transient performance of the engine when the sampling interval of the controller was varied. A two term (PI) controller algorithm was implemented in each case and with the engine running at the medium speed, medium load operating point step changes in voltage of both +0.6V(+150 rpm) and -0.6V(-150rpm) were injected into the desired speed channel of the microcomputer ADC.

Tests were conducted using the four sample intervals shown in Table 4 which also lists the corresponding controller parameters. The test results are shown in Figure 6.

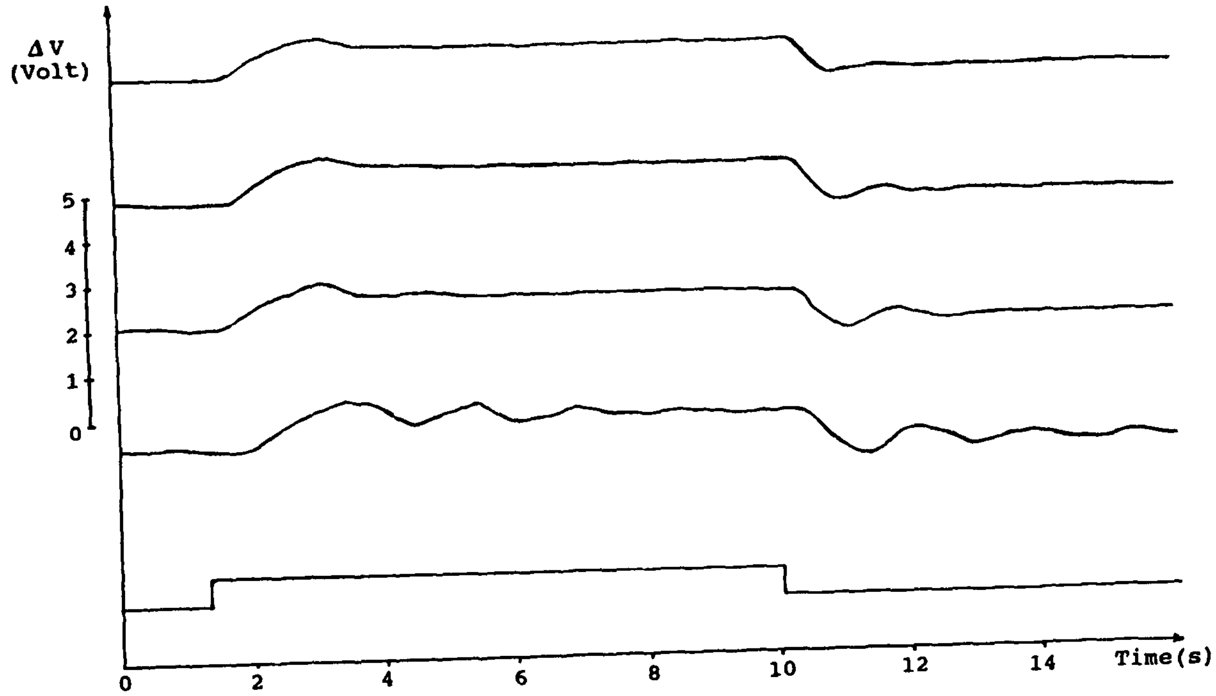


Figure 6 Effect of sampling interval on closed-loop response of diesel engine

Table 4 Selected sample intervals and corresponding controller parameters

Sample interval (s)	Controller parameters	
	K_p	K_i
0.07	4.5	0.421
0.15	4.5	0.904
0.3	4.5	1.807
0.5	4.5	3.012

The response obtained when T=0.07s and T=0.15s both exhibit small overshoots and settle quickly. They are both considered satisfactory. Furthermore, both sample intervals lie within the range $0.06 \le T \le 0.17$s suggested by the two rules of thumb. The effect of progressively increasing the sample interval first to T=0.3s and then to T=0.5s can be seen in each case to result in an increase in the oscillatory behaviour of the response and a consequent lengthening of the settling time. Taken as a whole, the results imply that a sample interval in the range $0.06 \le T \le 0.17$s is to be preferred and also that the system is reasonably insensitive to relatively small changes in T.

6. FUTURE DEVELOPMENTS

The controller trials outlined in this paper have been addressed simply to the problem of engine speed regulation, and in particular, regulation about one engine operating point using a single controller with fixed settings. The identification results confirm that such a controller cannot provide optimum performance for the engine over its full operating range. The changing dynamics of the engine make it particularly amenable to be controlled using self-tuning/adaptive techniques (17-20) and this is a specific area that RNEC intends to investigate further.

The work reported in this paper has been extended recently to include a study of both deadbeat and rule-based controllers together with the development of an on-line engine protection and monitoring facility (21). It is further intended that future work will extend the health monitoring facility to include a

diagnosis of diesel engine faults using on-line identification techniques.

7. CONCLUSIONS

The work discussed in this paper relates to the early stages of an investigation involving the identification and control of diesel engines used in ships of the Royal Navy. A number of areas of further study have been identified and the results reported in this paper indicate that a digital controller provides considerable scope for improving the overall performance of such engines.

8. REFERENCES

(1) D E Winterbone, C Thiruarooran, and P E Wellstead, "A Wholly Dynamic Model of a Turbocharged Diesel Engine for Transfer Function Evaluation", SAE Paper 770124, 1977.

(2) P E Wellstead, C Thiruarooran, and D E Winterbone, "Identification of a Turbo-Charged Diesel Engine", Proc 7th IFAC World Congress, Helsinki, Finland, 1978.

(3) D B Wellbourn, D K Roberts and R A Fuller "Proc I Mech E, 173, 1959, 575.

(4) D E Bowns, "The Dynamic Transfer Characteristics of Reciprocating Engines" Proc Inst.Mech.Engineers, 185, pp.185-201.

(5) J O Flower, P A Hazell and G P Windett, " Control Characteristics of Diesel Engines for Alternator Drives", Journal R.N.S.S., Vol 26, Nos. 1, 2, pp 3-12, 74-82, 1971.

(6) J O Flower and P A Hazell, "Sampled-Data Theory applied to the Modelling and Control Analysis of Compression Ignition Engines - Parts I and II", International Journal of Control, Vol 13, Nos. 3, 4, pp 549-562, 609-623, 1971.

(7) G E Harland, M A Pan'ko, K F Gill and J Schwarzenbach, "Pseudo-random signal testing applied to a Diesel engine". CONTROL February 1969.

(8) S A Billings, M B Fadzil, J L Sulley, and P M Johnson, "Identification of a Nonlinear Difference Equation Model of an Industrial Diesel Generator". Sheffield University Dept of Control Engineering Report, February 1986.

(9) C Thirurooran, "Modelling and Control of a Turbo-Charged Diesel Enginr", PhD Thesis, UMIST, 1979.

(10) D W Clarke, "A system identification package for VAX and LSI II computers", Report OUEL 1569/85, Dept of Eng Science, Oxford, UK, 1985.

(11) G M Van Der Mark, "Governors for Marine and Offshore Machinery", Trans. Inst. Marine Engineers, Vol 96, Paper 61, 1984.

(12) K J Astrom and B Wittenmark, "Computer Controller Systems, Theory and Design", Prentice-Hall, Inc., 1984.

(13) J G Zeigler and N B Nichols, "Optimum Settings for Automatic Controllers", Trans.American Soc. of Mech.Engineers, 1941.

(14) D W Clarke, "PID Algorithms and their Computer Implementation", Trans. Inst, M C, Vol 6, No 6, Oct-Dec, 1984.

(15) K Warwick and D Rees (Eds), "Industrial Digital Control Systems", Peter Perigrinus, 1986.

(16) J D Forrest, "Identification and Computer Control of a turbocharged diesel engine", MSc Dissertation, University of Sheffield, 1987.

(17) K F Gill, G E Harland and J Schwarzenbach, "Theoretical Design of an Adaptive Controller for an I.C.Engine. Control of Instrumentation, pp 50-53, October 1972.

(18) P M Zanker and P E Wellstead, "On Self-Tuning Diesel Engine Governors", Control Systems Centre Report 422, UMIST 1978

(19) P E Wellstead and P M Zanker, "Applications of Self-Tuning to Engine Control", In Harris, G J and Billings, S A (Eds), Theory and Applications of Self-Tuning and Adaptive Control, 1981.

(20) G E Harland and K F Gill, "Design of a Model - Reference Adaptive Control for an Internal Combustion Engine", Trans. Institute of Measurement and Control, Vol 6, pp 163-173, April 1973.

(21) N Mort and D C Langbridge, "Digital control and monitoring of a marine diesel engine", IMech E Seminar 1989-1 on the benefits of electronic control systems for IC engines, pp 33-40, 1989.

A Workable Dynamic Model for the Track Control of Ships

T.Holzhüter
Anschütz & Co., P.O.Box 1166, D-2300 Kiel 14, West Germany

1 Abstract

For the reliable installation of a track controller on different ships it is important that the controller is based on a simple model with parameters that are identifiable from the standard on board measurements. The usual linearized second order model of the hydrodynamic core of ship motion however is often ill conditioned with respect to identification. The reduction of the second order single input two output model having six coefficients to a first order model with only three coefficients is therefore considered. It can be justified in a theoretically satisfying way and also shows acceptable agreement with experimental results. The model is used in a commercially produced track controller that has proved high quality performance.

2 Introduction

For the installation of a track controller on seagoing ships, it is in general not feasible to perform extensive hydrodynamic investigations in order to establish a model for the ships' maneuvering characteristics. Therefore it is a great advantage if the controller relies on a model that can be estimated from the usual on board measurements of moderate accuracy. The usual linearized second order model for the description of ship hydrodynamics often is not well conditioned with respect to identification from such data.

This is partly due to the only approximate validity of the linearization which results in a dependence of the parameters on the specific maneuver. However, there are also general hydrodynamic reasons why the second order dynamics in the linear model usually are only weak or even negligible. When identifying a full second order model one is therefore close to a situation of overparametrization which may result in large and spurious variations of the estimated parameters.

In order to get a reliable performance of a track controller it is therefore advisable to use a reduced model the parameters of which can be estimated without danger of spurious fluctuations. The reduction of the second order single input two output model having six coefficients to a first order model with only three coefficients is therefore considered. It can be justified in a theoretically satisfying way and also shows acceptable agreement with experimental results. In fact part of this model has been used in autopilot design for course control of ships, see e.g. [1], [5].

We will discuss this reduction from a more system theoretic point of view. The resulting model has been used by other authors before, although without reference to the systems theory aspects. The distance of the second order model to the degenerate situation of negligible second order dynamics can be quantitatively characterized by the so called Hankel singular values. These quantities appear in the principal component analysis of linear systems which was introduced in [12]. It is closely connected the model reduction scheme via approximation of the Hankel operator of a system. Investigating the Hankel singular values of linear ship models shows that typically the distance from the degenerate situation is such that a reliable estimation of the second order dynamics from the usual on board measurements is not possible.

In principle the use of a simplified model in ship control increases the error of track prediction and thus would decrease the margins for save navigation. However, prediction of a track changing maneuver is generally difficult and in particular a prediction of turning circles is possible only with considerable errors, see e.g. [14]. Therefore we chose a different approach for the computation of ship maneuvers for our high precision track guidance system. We calculate the maneuvering trajectories of the ship according to the simplified model using a certain desirable time pattern of rudder motion. During the maneuver the control loop forces the ship to follow this trajectory using the theoretical rudder angle over time accompanied by some additional rudder action to compensate for the modelling errors and disturbances. This leads to good maneuvering performance and a high degree of predictability of the maneuvers as was reported in [4].

The paper is organized as follows. In sect.3 we briefly review the general features of a track control system for ships which establish the requirements for the ship model and the controller performance. In sect.4 we introduce the usual linear ship model and discuss the identification problems that have been observed with this second order model. The problems can be systematically understood by using the Hankel singular values of the system to characterize the distance of the model to a degenerate situation. For this purpose in sect.5 the corresponding theory of model reduction via Hankel approximation is briefly reviewed.

On this basis we apply the Hankel reduction scheme to a ship with very well know parameters in sect.6. In particular we discuss the dependence on scaling and the approximation error of the reduction process. This is completed in sect.7 by a discussion

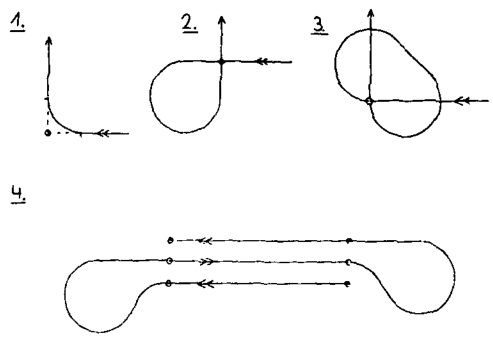


Figure 1: Track changing maneuvers

of selected ship models that we estimated from model tank experiments and from full scale sea trials. The identification problems using on board measurements can clearly be understood from this analysis and a simple reduced model of ship motion can be derived.

To illustrate the usefulness of the simplified ship model we finally report in sect.8.1 on its application in our model following track controller, especially in the construction of maneuvering trajectories, and present some practical experiences with our track control system.

3 General features of a ship guidance system

A ship guidance system generally consists of two blocks. the route management system and the track controller itself. The route management system includes editing and managing a numbered list of waypoints which constitute a route. At the waypoints a track changing maneuver has to be performed. In the widespread application as a low accuracy open sea track controller the track changing maneuver should be a smooth swing to the new track as is shown in fig.1. The radius of the turn should be selectable at the autopilot panel.

In special applications additional maneuvers are demanded. The track changes according to no.2 and 3 in fig.1 are for cases, in which the ship should follow the whole track including the waypoint. The long maneuver 3 is a solution for the case, where obstacles such as shallow water make the maneuver 2 impossible. In the case of dredging vessels and mine hunters several parallel track with a small distance are

typical. For this case the maneuver type 4 is useful.

For the simple open sea track controller which only allows maneuvers of type 1, a track change can be performed by switching over the calculation of the track error to the new track at an appropriate point. This point is depending on the desired radius of turn and the controller gains. If the controller has only small gains, this will result in a smooth swing to the new track.

For a high precision track control especially for the geometrically more complicated maneuvers 2,3,4 in fig.1 it is more adequate to calculate a whole maneuvering trajectory. The ship is then forced by the track controller to follow this reference trajectory. For the calculation a sufficiently accurate model of the ship is necessary. The typical track error for advanced track control is of the order of a few meters which determines the accuracy of the calculation as well as that of the position sensor.

Apart from the track changing maneuvers the controller has to manage the transition to the reference track from an arbitrary starting position and direction of the ship. Similarly the interruption and restarting of track control has to be possible. For this problem, it is suitable to develop a general program for the calculation of maneuvering trajectories which will be described in more detail in sect.8.2. For the construction of the trajectories a model of ship motion is necessary that captures the principal aspects of ship motion and is simple enough to be reliably estimated from standard measurements.

The use of precomputed maneuvering trajectories in a track control system can be seen as an alternative way of performing accurate track predictions. For save navigation it is desirable to predict the path of the ship during course changing maneuvers. However the maneuvering behavior of a ship is difficult to compute due to nonlinearities and is subject to considerable disturbances due to weather and loading conditions. A compromise between model complexity and the necessities of parameter identifiability has been attempted in [16]. However at least for a commercial implementation of a track controller the model presented there still seems too complex.

Instead of implementing a large model which necessarily includes a large number of unknown coefficients one practical approach is to adapt a simple ship model to the changing dynamic behavior of the ship by a parameter estimation scheme. This approach was presented in [14]. The adaptation mechanism has the disadvantage of being a dangerous part in an on line control device, since it introduces a severe nonlinearity in the whole algorithm.

Our approach instead uses a very simple model of the ship dynamics as well to compute the maneuver and afterwards forces the ship to the reference path. It uses real feedback applied to the ship by the controller instead of feedback to the prediction model for the ship path from an identification scheme as in the approach [14]. This leads to less steady rudder action, but the result is a very accurately known ship path. From our experience the use of the simple ship model in the trajectory computation

results in rudder action that is to a certain extent induced by modelling errors but still is acceptable from the navigational aspects.

4 Problems in the modelling of ship motion

4.1 Dynamic ship model

For the design of a track controller a model is necessary that describes all relevant aspects of ship motion. On the other hand it has to be simple enough, such that its parameters may be identified from the standard on board measurements in a reliable way. In addition possible nonlinearities have to be manageable for the controller design.

Linearity of the controller will be achieved by linearization of ship motion along a reference trajectory. Fig.2 shows the different coordinate systems involved in the description the ship motion. For simplicity of the equations it is appropriate to use a coordinate system that is tangent to the reference trajectory at the point that is closest to the actual position. If the track errors are small, this point is unique. In the following it is usually assumed that the position and the heading are transformed to this coordinate system.

For computational convenience the local coordinate system is shifted and rotated from time step to time step in such a way that it is always tangent to the reference trajectory and that the x-coordinate always represents the arc length along the trajectory. Thus on a straight course, the local coordinate system remains fixed. However during maneuvers, its origin is moving on the involute curve of the reference trajectory.

The linearization of the hydrodynamic equations of motion of a ship leads to a core model of dimension two in the state variables rate of turn r and sway velocity v, see [8]. With definitions from fig.2 we have the following state space model

$$\dot{\begin{pmatrix} r \\ v \end{pmatrix}} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} r \\ v \end{pmatrix} + \begin{pmatrix} b_{11} \\ b_{21} \end{pmatrix} \delta \tag{1}$$

This model however often is ill conditioned with respect to identification when using on board measurements. We will discuss this point in sect.4.2. The remaining states can be derived from kinematic relations according to fig.2. Assuming the presence of an additional stationary current (d_x, d_y), we get the equations

$$\dot{\psi} = r \tag{2}$$

$$\dot{x} = U \cos\psi + v \sin\psi + d_x \tag{3}$$

$$\dot{y} = U \sin\psi + v \cos\psi + d_y \tag{4}$$

Here ψ is the heading angle and (x, y) are the two components of the position of the ship in the local coordinate system. The longitudinal velocity U of the ship is measured

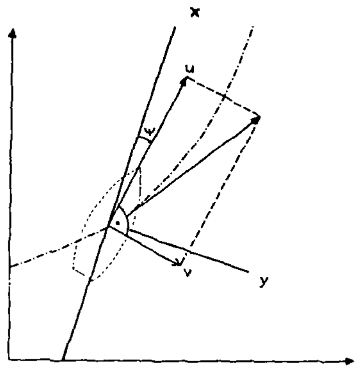


Figure 2: *Coordinate systems for ship model*

by the speed log and thus can be viewed as a known parameter. During track control, the relative heading angle ψ is small and thus the equ.(3),(4) can be linearized

$$\dot{x} = U + d_x \quad (5)$$
$$\dot{y} = U\psi + v + d_y \quad (6)$$

However for the computation of the maneuvering trajectories (*see sect.8.2*) or in the extrapolation step of the Kalman filter (see sect.8.1) the errors in this linearization turn out to be too large. Therefore this totally linearized model is only used for the calculation of the Kalman filter and controller gains.

The drift in the longitudinal direction d_x does not influence control directly. However its estimation is necessary to improve the overall controller performance. After course changes of about 90° it will appear as a cross drift and thus becomes important for control. Therefore its permanent estimation leads to less track deviations during track changes.

4.2 Identification problems with the ship model

Although a second order model like (1) seems to be rather simple it still can pose problems in parameter identification. This is particularly true using on board measurements, where the accuracy of the position measurements is often rather limited.

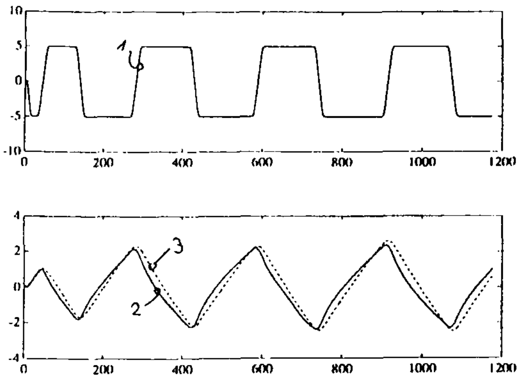


Figure 3: Normalized Variables δ [1], r [2], and $-v$ [3] in a standard maneuver.

When both variables v and r are measured, the identification of the six parameters in the discretized version of model (1) is possible by interpreting the model equations as two simultaneous least square problems. For simplicity we use the same symbols for the parameters as in (1).

$$(7) \qquad r_{t+1} = (\hat{a}_{11} \quad \hat{a}_{12} \quad \hat{b}_{11})(r_t \quad v_t \quad \delta_t)^T + \hat{e}_{r,t}$$

$$(8) \qquad v_{t+1} = (\hat{a}_{21} \quad \hat{a}_{22} \quad \hat{b}_{21})(r_t \quad v_t \quad \delta_t)^T + \hat{e}_{v,t}$$

Typically the two regression variables r and v are highly correlated and the estimation in both problems may become ill conditioned. This has been reported from an investigation on the identifiability of the hydrodynamic coefficients in typical maneuvers where a simultaneous drift of two of the hydrodynamic coefficients was observed. In [6] this is termed a cancellation effect, since the cause for the parameter fluctuations is the cancellation of the regression influences of r and v due to their correlation.

This can be understood when looking at the time series of the state variables r and v in a typical test maneuver like a z-maneuver as shown in fig.3. The yaw rate r and the negative sway velocity v nearly move in parallel, as is emphasized in fig.3 by an appropriate choice of scales. It is obvious that identification of the second order dynamics for such a system is at least difficult.

In [6] it is argued from the hydrodynamic theory of slender bodies that a tendency towards the occurrence of this cancellation effect is inherent in ship steering dynamics. Therefore identification of a model structure like (1) can pose problems and the pa-

rameters may be partly random. However the deeper system theoretic aspects of this phenomenon have not been considered in [6].

To illustrate the system theoretic causes of this cancellation effect, we will consider the extreme case of perfect correlation, when the variables r and v are equal up to a scaling factor. We consider the special case $v = r$ only, since we can always transform to this situation by choosing appropriate units of measurement. Thus we assume the data generating model:

$$\begin{pmatrix} r \\ v \end{pmatrix}_{t+1} = \begin{pmatrix} a & 0 \\ 0 & a \end{pmatrix} \begin{pmatrix} r \\ v \end{pmatrix}_t + \begin{pmatrix} b \\ b \end{pmatrix} \delta_t \tag{9}$$

The estimation asymptotically is governed by the information matrix of the regressors, see e.g. [11]. If we restrict our attention to the estimation of the system matrix (a_{ij}) and thus to the regression vector $(r \quad v)^T$ the information matrix for a time serie of length N will be singular.

$$\operatorname{cov}(\hat{a}_{i1}, \hat{a}_{i2})^{-1} = \frac{1}{N} \begin{pmatrix} E(r^2) & E(rv) \\ E(vr) & E(v^2) \end{pmatrix} = \frac{1}{N} E(r^2) \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \tag{10}$$

$$= \frac{1}{N} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 2E(r^2) & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}^T \tag{11}$$

As can be seen from the eigenvector decomposition (11), only the sum of the parameters $(\hat{a}_{i1} + \hat{a}_{i2})$ can be determined. As the asymptotic estimate for the discrete time system matrix we may thus get any matrix of the form

$$A = \begin{pmatrix} (a - \epsilon) & \epsilon \\ \delta & (a - \delta) \end{pmatrix} \tag{12}$$

where the values of ϵ and δ are completely arbitrary. It can easily be seen that this model has one eigenvalue at a and an uncontrollable eigenvalue that depends on ϵ and δ. We will see in sect.6 that the identification problems in fact originate from the point that typically the linear model for ships is close to the non-minimal, uncontrollable system (9). The model structure (1) is thus close to a situation of over-parametrization leading to the joint randomness of some of the parameters. In [15] parameter tracking results in the context of an autopilot design using the full second order model have been presented. Although it is not mentioned there, the simultaneous parameter fluctuations due to a temporary loss of identifiability seem to be present there also.

In [6] it is proposed to overcome this problem by introducing a nonlinear structure into the model. In the context of linear modelling for control purposes however it seems more advisable to reduce the system to a first order model, thus getting a simpler model, based on parameters that can be reliably estimated from typical maneuvers. The model reduction can be performed by using truncated balanced realizations. We will discuss the reduction of ship models via this route in sect.6 and 7.

5 Model reduction via Hankel approximation

The model reduction via approximation of the Hankel operator of a system is closely connected with the so called balanced realization of a linear system. The balanced realization of a stable state space model is a special canonical form

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{13}$$
$$y(t) = Cx(t) \tag{14}$$

for which the controllability and observability gramians G_c and G_o, defined by

$$G_c = \int_0^\infty e^{\tau A} BB^T e^{\tau A^T} d\tau \tag{15}$$
$$G_o = \int_0^\infty e^{\tau A^T} C^T C e^{\tau A} d\tau \tag{16}$$

have diagonal form and are equal, see [2] for details. The gramians are solutions of the algebraic Lyapunov equations

$$AG_c + G_c A^T = -BB^T \tag{17}$$
$$A^T G_o + G_o A = -C^T C \tag{18}$$

The eigenvalues of the product $G_c G_o$ are invariants of the system. The square roots of these eigenvalues are called Hankel singular values σ_i since they are identical with the singular values of the Hankel operator of the linear system (13).

$$\sigma_i = \lambda_i (G_c G_o)^{1/2} \tag{19}$$

If the diagonalization of the gramians is carried out in such a way that the singular values in the diagonals $\{\sigma_k; k = 1 \ldots n\}$ appear in decreasing order, the σ_k can be thought of as describing the relative importance of system dynamics of order $j \leq k$ for the output of the system. The largest singular value σ_1 establishes the Hankel norm of the system which characterizes the maximum system gain. A numerically reliable algorithm for balancing of linear systems was presented in [9].

Balanced realizations have the interesting property that every subsystem generated by omitting the last m states from the system leads to a meaningful approximation of the original system. By partitioning the state vector of the balanced realization after the first m states according to

$$\dot{\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} B_1 \\ B_2 \end{pmatrix} u \tag{20}$$
$$y = \begin{pmatrix} C_1 & C_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{21}$$

one may obtain the stable truncated system

$$\dot{x}_1 = A_{11}x_1 + B_1u \tag{22}$$
$$y = C_1x_1 \tag{23}$$

The reduction process can be interpreted as omitting the least important states from the system. In general, it is possible to derive approximations that are better in the sense of the Hankel norm than the truncated realizations but the latter are more simple to compute. The distance in Hankel norm of any approximation from the original system is bounded from below by the largest neglected Hankel singular value σ_{m+1}.

In the context of control design the accuracy of an approximating system $\hat{H}(s)$ is suitably described by the H^∞-norm of the error transfer function. If $H(s)$ is the transfer function corresponding to the original system (20), (21) it can be shown that the norm of the approximation error of the truncated system (22),(23) is bounded by twice the sum of the neglected singular values.

$$\left|\hat{H}(s) - H(s)\right|_{H^\infty} \leq 2 \sum_{i=m+1}^{n} \sigma_i \tag{24}$$

Balanced realizations only exist for stable systems. A reduction of unstable systems via balanced realizations is possible by decomposing the transfer function into a sum of a stable and an unstable part. Only the stable system has to be reduced while the unstable part should remain unchanged, see [2].

Using this technique it would be possible to reduce the order of the full unstable ship model constituted by (1),(2),(6). However, the states r and v have a very significant physical meaning in the ship control problem and have to be preserved in the reduction process. This would not be the case when reducing the full model which will result in mixed states. Therefore we will perform the reduction process for the stable core model (1) viewed as a separate system, although no guaranteed error bounds will be available for the reduced unstable system in that case.

In our example the model reduction via balanced realization can provide a first order model as the main principal component of the system dynamics. A quantitative measure for the deviation of the second order model from the first order approximation is provided by the ratio of the hankel singular values.

The core model of ship motion (1) was discussed to be close to an uncontrollable model as described by (9). For this degenerate model, one of the Hankel singular values is exactly zero, as can directly verified by solving the Lyapunov equations (17),(18).

$$G_c = \begin{pmatrix} -\frac{b^2}{2a} & -\frac{b^2}{2a} \\ -\frac{b^2}{2a} & -\frac{b^2}{2a} \end{pmatrix} \tag{25}$$

$$G_o = \begin{pmatrix} -\frac{1}{2a} & 0 \\ 0 & -\frac{1}{2a} \end{pmatrix} \tag{26}$$

$$G_c G_o = \frac{b^2}{4a^2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \tag{27}$$

The singular values of the matrix $G_c G_o$ according to (19) are the Hankel singular values of the degenerate system (9).

$$\sigma_{1,2} = \begin{cases} \frac{b^2}{2a^2} \\ 0 \end{cases} \tag{28}$$

This corresponds to the uncontrollability of the degenerate system. The second state in the balanced realization can be omitted without loosing anything of the system dynamics. The degenerate system (9) actually is only a first order SIMO model and in continuous time can be written as:

$$\begin{aligned} \dot{r} &= -\frac{1}{T} r + \frac{K}{T} \delta \\ \begin{pmatrix} r \\ v \end{pmatrix} &= \begin{pmatrix} 1 \\ \alpha \end{pmatrix} x \end{aligned} \tag{29}$$

Since it can be reliably estimated from the usual on board measurements, we used this reduced model of ship steering dynamics in our track control system for controller design and for the computation of reference trajectories. We will discuss the approximation error introduced by this simplification in the following sections.

6 Hankel reduction for a typical ship model

We will now discuss the reduction of the usual second order model of coupled yaw and sway motion of the ship as described by (1) to a first order model via truncated balanced realizations. The reduced model structure has been used on pragmatic grounds before, especially for the SISO-System with output r only. In that case it is known as the Nomoto model, referring to [13].

We will consider the reduction process for the SIMO-model with both outputs $(\, r \quad v\,)^T$ for a special ship for which reliable parameter values are available. We select the well known Mariner class ship which has been investigated by numerous institutions and authors both in model tests and in full scale trials. The parameters for the linear model (1) are given in [7] using the Prime notation in which the variables are scaled with the ship speed U and length L.

$$t^* = t\,(U/L) \quad v^* = v\,(1/U) \quad r^* = r\,(L/U) \tag{30}$$

Table 1: Normalized parameters for the Mariner class ship

T_1	$= 5.660$	T_2	$= 0.372$
K_r	$= 3.855$	K_v	$= -1.898$
T_{3r}	$= 0.889$	T_{3v}	$= 0.189$

Introducing these variables and omitting the star superscripts in the following we obtain the model:

$$\frac{d}{dt}\begin{pmatrix} r \\ v \end{pmatrix} = \begin{pmatrix} -2.093 & -3.394 \\ -0.335 & -0.770 \end{pmatrix}\begin{pmatrix} r \\ v \end{pmatrix} + \begin{pmatrix} 1.627 \\ -0.170 \end{pmatrix}\delta \tag{31}$$

The parameters of the corresponding transfer functions from rudder δ to r and v are given in table 1 using the normalized time scale according to (30).

$$H_{\delta r}(s) = \frac{K_r(1 + sT_{3r})}{(1 + sT_1)(1 + sT_2)} \tag{32}$$

$$H_{\delta v}(s) = \frac{K_v(1 + sT_{3v})}{(1 + sT_1)(1 + sT_2)} \tag{33}$$

It was mentioned in sect.5 that the scaling of the different input or output variables is important for the Hankel reduction process. In our case of a system with one input and two outputs, only the relative scaling of the output channels is important for the reduction process. We introduce an additional scaling factor c into the omitted unity output matrix of system (31) which modifies the units of measurement for v.

$$\begin{pmatrix} r \\ \hat{v} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & c \end{pmatrix}\begin{pmatrix} r \\ v \end{pmatrix} \tag{34}$$

This model can be reduced to a first order model by truncating its balanced realization.

$$\dot{x} = -\frac{1}{T}x + \frac{K}{T}\delta \tag{35}$$

$$\begin{pmatrix} r \\ \hat{v} \end{pmatrix} = \begin{pmatrix} 1 \\ c\alpha \end{pmatrix} x \tag{36}$$

We kept the scaling factor c to obtain parameters that are independent of scaling. The parameters T and K are those of the transfer functions:

$$\hat{H}_{\delta r}(s) = \frac{K}{(1 + sT)} \tag{37}$$

$$\hat{H}_{\delta \hat{v}}(s) = \frac{K\alpha c}{(1 + sT)} \tag{38}$$

The reduced transfer function (37) is well known in ship control as the Nomoto model. The approximation of sway velocity as a fixed multiple of yaw rate according to (38) has also been used as an approximation in the ship modelling context, see e.g. [10],[14].

Table 2 shows the variation of the parameters $\{T,\ K,\ \alpha\}$ of the reduced model as a function of the scaling factor c. It can be seen that the parameter α is nearly independent of scaling. However the two other parameters vary in a certain range. For extreme scalings there is an asymptotic value for both. The two asymptotic scaling situations correspond to the reduction of the two channels of model (31) viewed as independent SISO systems. For small values of c (which correspond to a large unit of measurement for v) the approximation error in the second channel is numerically irrelevant when compared to the first channel. Therefore the reduction process virtually is the reduction of the SISO system with input δ and output r. For large values of c we have the corresponding result for the second channel.

The dependence of the model reduction for the SIMO model on the relative scaling makes the reduction process appear somewhat arbitrary. It should however be noted that the two channels of system (1) are not just two independent outputs but are related in the full ship model. If we use the natural scaling according to (30), the full model equations (2),(6) form the sum $z = (\int r\ dt + v)$. Thus there is a connection between the scaling of the two error signals of r and v. The corresponding reduction of the model (31) with the sum $\dot{z} = (r + \dot{v})$ as output variable leads to parameters $\{T, K\}$ and Hankel singular values close to the situation with $c = 1$, as can be seen from table 2. Thus the natural scaling using the units from (30) seems to be adequate also for the model reduction problem.

The ratio of the Hankel singular values shown in table 2 provides a quality measure of the approximation, since according to (24) the error norm is bounded by the second singular value and the system gain is characterized by the largest singular value. Inspection of column 5 of table 2 shows that the reduction of the yaw channel ($c \to 0$) to a first order model obviously is less accurate than for the sway channel ($c \to \infty$). This will also turn out in the experiment data presented in sect.7.

It was mentioned in sect.4.1 that the second order ship model (1) often is close to the degenerate system (9). It was shown in sect.5 that this situation is be characterized by one of the Hankel singular values being exactly zero. It reflects that the second order dynamics of the degenerate model is totally negligible. As can be seen from table 2 the Mariner ship for the natural scaling situation ($c = 1$) has a ratio of the singular values of 0.075. This can be interpreted to indicate that the second order dynamics only account for less than 10% of the system gain, see [2]. Thus especially for low accuracy data from on board measurements it cannot be expected that the second order model (1) can be discriminated against the degenerate model (9). Therefore the use of the reduced model for identification is advisable.

Using the second singular value from table 2 it can be seen that both static gains will only have an approximation error of well below 10%. This can be verified by comparing with the values for K_r and K_v given in table 1.

$$(39) \qquad |K_r - K| = \left|H_r(0) - \hat{H}(0)\right| \leq \left|H_r(s) - \hat{H}_r(s)\right|_{H\infty} \leq 2\sigma_{2r} = 0.282$$

$$(40) \qquad |K_v - K\alpha| = \left|H_v(0) - \hat{H}(0)\right| \leq \left|H_v(s) - \hat{H}_v(s)\right|_{H\infty} \leq 2\sigma_{2v} = 0.051$$

The typical variations of the static gains when using data from different maneuvers of a ship usually are much higher. Thus the reduction process will not introduce significant additional errors into the model. On the contrary, it turned out that the parameters of the reduced model (29) constitute the reproducible part of the ship model while the second order dynamics inherent in the full model (1) to a large extent depend on the specific experimental conditions.

As an alternative to the reduction of the SIMO system (1) we could view the system as a vector of two independent second order transfer functions. Since in the structure (1) these transfer functions have a common denominator by definition, only two Hankel singular values in the artificial non-minimal forth order model would be distinct from zero. Generally this would not be the case for transfer functions derived from experimental data.

The two second order SISO models may be reduced independently via balanced realizations. Scaling does not influence the reduction process in that case since we only deal with SISO systems. Each of the resulting approximations are equal to one of the first order models obtained from the asymptotic scalings already discussed. The two first order models afterwards can be recombined into a diagonal second order model.

$$(41) \qquad \dot{\begin{pmatrix} r \\ v \end{pmatrix}} = \begin{pmatrix} -0.2491 & 0 \\ 0 & -0.1580 \end{pmatrix} \begin{pmatrix} r \\ v \end{pmatrix} + \begin{pmatrix} 0.8901 \\ -0.6964 \end{pmatrix} \delta$$

This model (41) contains less coefficients than (31) although it is still of order two. Since each channel is separate it will show no ill conditioning in identification as discussed in sect.4.2. The subsequent reduction of this model to a first order model again depends on scaling. The results are very close to those for the direct reduction shown in table 2 and are omitted. In principle the reduction process depends on the sequence of model structures followed in the reduction process. In conclusion it can be seen from this example that the ship model typically is not identical to the degenerate model (29). However, the distance from that situation is such that with measurements of low accuracy it cannot be expected to discriminate against the simplified model. We will pursue this point in the next section by actually using on board measurements of moderate accuracy for the identification of the ship model.

Table 2: Scaling dependence of parameters of the reduced model

c	T	K	α	σ_2/σ_1	σ_2
0.01	4.014	3.573	−0.4307	0.0788	0.1409
0.1	4.019	3.575	−0.4307	0.0788	0.1411
1	4.415	3.733	−0.4333	0.0751	0.1528
10	6.225	4.373	−0.4419	0.0314	0.3112
100	6.327	4.406	−0.4423	0.0262	2.557
1000	6.328	4.407	−0.4423	0.0262	2.551
$\dot{v} + r$	4.780	3.703		0.03914	0.0758

Table 3: Conditions for the experiments

Experiment No.	L [m]	U [m/sec]	δ_{max} [degr]	
1	47	3.1	5	model
2	47	6.2	5	model
3	54	3.2	20	full scale
4	54	6.2	20	full scale

7 Ship models from experimental data

The problems of modelling ship motion according to the standard structure (1) will now be illustrated using parameter estimation results from experimental data. We will analyze data from model tank experiments and from full scale sea trials. Unfortunately up to now we did not have data referring to the same ship as a model and as full scale ship. Some characteristics of the experiments are shown in table 3. The data concerning the model experiment were transformed to the full scale ship using Froudes law of similarity in order to get comparable conditions. The angle δ_{max} is the rudder input used in the z-maneuver, see fig.3.

Data from the model tank experiments are much more accurate and the estimation of the hydrodynamic core model (1) according to the least square approach of (7),(8) in general does not pose problems. Using the data from on board measurements however, the estimation results in some cases are unusable which is mainly due to the moderate accuracy of the position measurement.

Parameter estimation results are compiled in table 4. The parameters for the full model are given as coefficients of the transfer functions (32), (33). The parameters of the reduced model are also given according to (35), (36) using the natural scaling $c = 1$ in the reduction process.

For the model experiments the ratio of Hankel singular values is $\sigma_2/\sigma_1 \approx 0.04$ which is close to the value that appeared in 6. This ratio corresponds to a clear separation of the two time constants of the second order model. The experiments with the full scale ship in case 3 produced a pair of poles close together and accordingly a ratio $\sigma_2/\sigma_1 = 0.15$. For experiment 4 the estimation even produces a pair of complex conjugate poles and singular values of even more similar size: $\sigma_2/\sigma_1 = 0.30$.

The diverging quality of the two regression equations (7) and (8) using on board measurements can already be deduced from the variance of the residues in the regression. In table 4 the signal to noise ratios γ calculated from the residual variance are given.

$$\gamma_r = \frac{\hat{\sigma}_e^{(r)}}{\sqrt{\mathrm{Var}(r)}} \tag{42}$$

$$\gamma_v = \frac{\hat{\sigma}_e^{(v)}}{\sqrt{\mathrm{Var}(v)}} \tag{43}$$

The low signal to noise ratio in the regression for the sway velocity v can be attributed to the low accuracy of the position information from which the time serie of v is calculated. Therefore a separated ARX(1) estimation for the yaw rate r corresponding to the model (35) resulting in estimates for T and K and an additional regression of v on r to estimate the parameter α according to (36) will give more reliable and

Table 4: Identification results for the standard model (1) and for the reduced model (29)

Experiment No.	1	2	3	4
T_1	5.44	4.34	3.70	1.13
T_2	0.619	0.470	1.23	
K_r	21.8	17.1	1.11	0.683
T_r	0.881	0.686	6.51	3.44
K_v	−12.2	−8.76	−8.07	−3.22
T_v	0.415	0.377	0.847	0.919
K	21.8	16.9	2.40	1.62
T	4.98	3.87	4.38	3.26
α	−0.518	−0.478	−3.39	−2.53
σ_2/σ_1	0.039	0.038	0.15	0.30
γ_r	200	325	62.4	66.8
γ_v	131	231	16.9	24.6

reproducible results under these circumstances. As was already stated we followed this route in the design of our track control system, see [4].

8 Experiences with a track controller for ships

8.1 General controller structure

The controller has to meet two different performance requirements. It has to provide track keeping and track changing capability. Track changing is reduced to a track keeping problem due to the precomputation of the reference trajectory as will be described in sect.8.2. A theoretical rudder pattern is chosen for the construction of the trajectory. Thus it is rather straight forward to use this known control input for feed forward control. Because of parameter uncertainties and unmeasured disturbances the ship will move in a different way as is predicted from the trajectory computation. For compensation, the track keeping loop has to provide the correcting rudder input. This feedback part of the controller can be the same as the track keeping controller, since linearization of the ship model at low yaw rates is approximately the same as at yaw rate zero. The whole system is thus a combination of feed forward and feedback control.

Since not all states are measurable and the measurements are corrupted by noise, a state estimate $\hat{x}$ has to be reconstructed with a Kalman filter. The implementation of the Kalman filter is in discrete time. From the system of equations (29),(2),(3), and (4) we have the state vector

$$x = (\,r \quad \psi \quad x \quad y \quad d_x \quad d_y\,)^T$$

the measurement vector $z = (\,\psi_m \quad x_m \quad y_m\,)^T$ and the control variable $u = \delta$. This model corresponds to a nonlinear discrete time state space model:

$$x_{t+1} = f(x_t) + Gu_t + w_t \qquad (44)$$

$$z_t = Hx_t + v_t \qquad (45)$$

Here v_t and w_t are the measurement and the process noise respectively, both assumed to be white. For a more detailed discussion of the ship controller design along the principles of LQG control we refer to [3]. The whole controller is described by the equations:

$$\hat{x}^*_{t+1} = f(\hat{x}_t) + Gu_t \qquad (46)$$

$$\hat{x}_{t+1} = \hat{x}^*_{t+1} + K(z_t - H\hat{x}_t) \qquad (47)$$

$$\hat{u}_t = L(\hat{x}_t - \bar{x}_t) \qquad (48)$$

The geometric nonlinearity of the system according to (3),(4) is preserved in the extrapolation step (46) of the Kalman filter, $\hat{x}^*$ denoting the extrapolated state estimate.

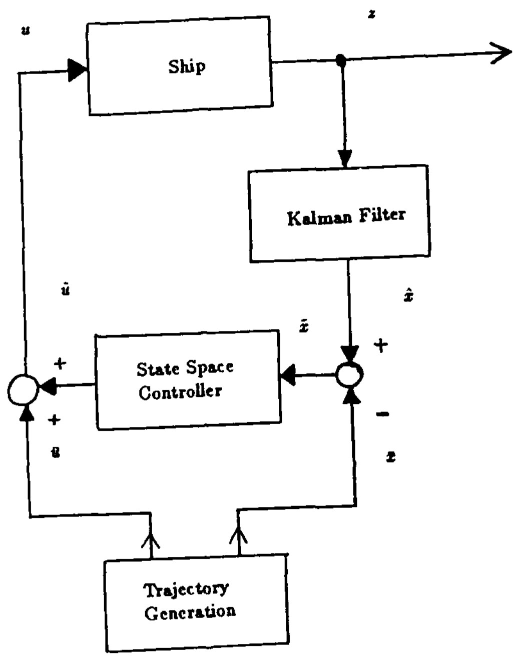


Figure 4: Model following control

For the calculation of the Kalman gains K and the controller gains L the linear approximation according to (29),(2),(6) is used, see [4] for more details.

The feedback control $\hat{u}$ in (48) is based on the differences of the estimated states $\hat{x}$ from the reference state vector $\bar{x}$ as obtained from the trajectory construction. The total control signal u is the sum of the feedforward control $\bar{u}$ and the correction $\hat{u}$ resulting from the feedback loop.

$$u_t = \hat{u}_t + \bar{u}_t \tag{49}$$

The information flow for this control structure is summarized in fig.4. It can be viewed as a generalization of the well know method of prefiltering the reference signal in linear control theory and can be coined model following control.

8.2 Maneuvering trajectories

The planning of the track changing maneuvers for model following control has to take into account the equations of motion of the ship. It was discussed in sect.3 that different sorts of maneuvers have to be available for track changes. The simple construction consisting only of circular arcs and straight lines is geometrically easy to solve, however, it results in a very unsteady maneuvering. The rate of turn at the junction of a circle with radius R and a straight line has to jump from zero to a value $\bar{r} = U/R$ resulting in large rudder angles at that point.

On the other hand, the construction of arbitrary trajectories which obey the equations (29),(2),(4) and fulfill certain given boundary conditions is very hard to solve. Therefore as a compromise we added to our trajectory construction tool box two further basic elements complementing circles and straight lines. These two elements model the start of turn and the stop of turn phase. They each correspond to a certain rudder step response. The two elements are used as intermediate elements at the junction of circles and straight lines.

The resulting trajectories have a time pattern of the rudder angle as is shown in fig.5. A certain initial rudder is used on the intermediate element until the desired value of the rate of turn is reached. Then the rudder is reduced to the steady state value on the circular arc. At the end of the turn a certain counter rudder is used to stop the turn. These trajectories can be constructed in an iterative procedure. As a first approximation, a trajectory consisting of circular arcs and straight lines alone is constructed. This basic geometric construction is iteratively modified according to the needs introduced by the intermediate elements. The computational burden for this procedure is reasonable.

The approach of a track at the beginning or after interruption of track control is also solved in this manner. A general and flexible on line program was developed. It allows to compute suitable trajectories starting from arbitrary prescribed points and

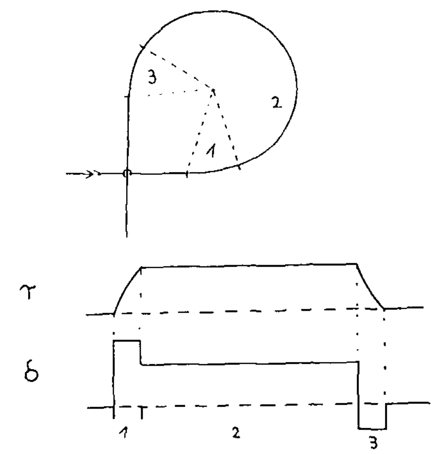


Figure 5: Typical trajectory construction

directions and ending at prescribed points and directions. A set of priorities can be chosen to influence the qualitative track pattern.

8.3 Practical experiences

The track controller is implemented on a Motorola 68000 microprocessor as part of a general autopilot and steering control system. The control software was written is Pascal. Apart from track control the usual autopilot functions course control and rate of turn control are available.

Several full scale trials have been performed, during which the track controller showed good steering performance. In fig.6 an example of a trial is shown including three different types of maneuvers. The ship speed was 3 m/sec. As a position reference system the radio navigation system Syledis was used. It has statistical position variations of about 3 m, however there may be systematic shifts in the position up to about 30 m.

Because of lack of an absolute position reference the track error had to be calculated as the distance of the measured position from the reference trajectory. During track keeping phases the error is well below 5 m. During track changes the error was larger. This was due to the use of a preliminary ship model, for which the coefficient α in

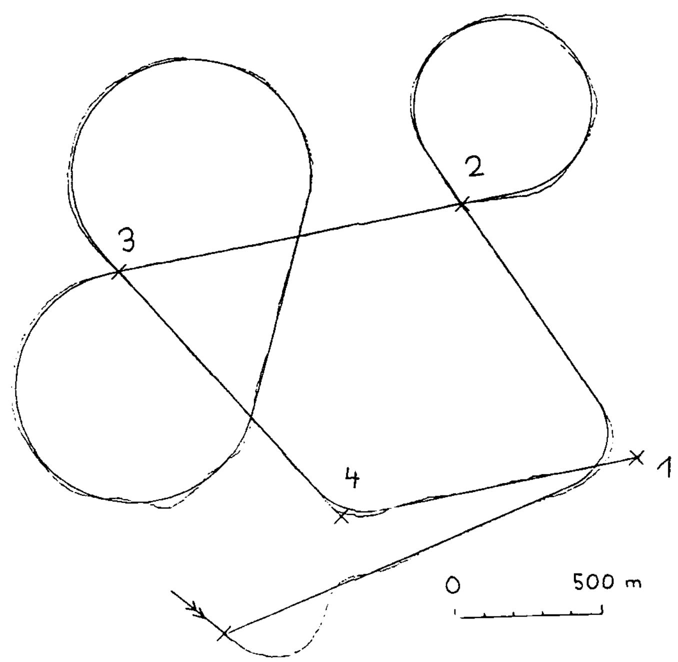


Figure 6: Full scale trial

(29) was taken too small. This resulted in some transient behavior mainly at the start and stop phases of the turning maneuvers. The trial was performed when a current of about 1m/sec was present. It can be seen that this does not seriously affect the track keeping performance.

9 Conclusions

The modelling of ship motion using a simple, mainly linear model is sufficient to get satisfying track control of a ship. It has the great advantage that its parameters can be identified from on board measurements which is quite important for an industrial application.

With an appropriate position reference system a track error of a few meters turns out to be obtainable. In addition the computation of a reference trajectory for the ship maneuvers can provide a efficient way to get very accurate track predictions. Even when using only a strongly simplified model, the modelling errors can be eliminated by the control loop which forces the ship to follow the reference trajectory. The improved predictability of the ship track is an important aid for performing save course changing maneuvers.

References

[1] J.v.Amerongen; Adaptive steering of ships - A model reference approach; Automatica, Vol.20, No.1, pp.3-14, 1984.

[2] K.Glover; All optimal Hankel-norm approximations of linear multivariable systems and their L^{∞}-error bounds; Int.J.Control Vol.39 No.6 (1984) pp.1115-1193.

[3] M.J.Grimble, M.R.Katebi; LQG-design of ship steering control systems; in: Thoma (Ed.); Signal Processing for Control; Springer Verlag (1986).

[4] T.Holzhüter; A high precision track controller for ships; Paper accepted for presentation at 11.IFAC World Congress, Tallin, Estonian SSR (1990).

[5] T.Holzhüter und H.Strauch; An adaptive autopilot for ships: design and operational experience; Proceedings of the 10.IFAC World Congress, Munich (1987), pp.226-231.

[6] W.-Y.Hwang; Cancellation effect and parameter identifiability of ship steering dynamics; Int.Shipbuilding Progr. Vol.26 No.332 (1982) pp.90-102.

[7] C.G.Källström and K.J.Åström; Experiences of system identification applied to ship steering; Automatica Vol.17 No.1 (1981) pp.187-198.

[8] C.G.Källström; Identification and Adaptive Control applied to Ship Steering; PhD.Thesis Lund Institute of Technology; (1982).

[9] A.J.Laub, M.T.Heath, C.C.Paige, R.C.Ward; Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms; IEEE Trans.Automatic Control Vol.AC-32 No.2 (1987) pp.115-122.

[10] G.van Leuwen; A simplified non-linear model of a manoeuvring ship; Shipbuilding Laboratory, Delft University of Technology, Report no.262 (1970). ,

[11] L.Ljung; System Identification - Theory for the User; Prentice Hall (1987).

[12] B.C.Moore; Principal component analysis in linear systems: controllability, observability and model reduction; IEEE Tr. Automatic Control Vol. AC-26 No.1 (1981) pp.17-32.

[13] K.Nomoto et.al.; On the steering quality of ships; Int.Shipbuilding Progr.Vol.4 No.35 (1957) pp.354-370.

[14] P.O.Passenier, J.v.Amerongen; An adaptive track predictor for ships; Proceedings of the 8.Ship Control Symposium, The Hague, Netherlands (1987) pp.2.14-2.32.

[15] S.Saelid, T.Svanes, T.Onshus, N.A.Jensen; Design considerations, analysis and practical experience with an adaptive ships autopilot; 9.IFAC World Congress; Budapest, Hungary (1984).

[16] C.de Wit; An adequate and yet workable dynamic model to support track keeping measurements; Proceedings of the 8.Ship Control Symposium, The Hague, Netherlands (1987) pp.2.183-2.208.

ATTAINABLE STOPPING PERFORMANCE IMPROVEMENTS FOR GAS TURBINE/CPP SHIPS USING COORDINATED CONTROL OF POWER AND PITCH

by Larry C. Carroll
PDI CORP.

1. ABSTRACT

A propulsion dynamics investigation of a notional 8500 ton destroyer driven by two controllable pitch propellers (CPP) is presented. Operation with one and two engines per shaft modes is analyzed. The study focuses on the coordinated control of engine power and propeller pitch commands to produce the fastest realizable ship stopping performance without violating predefined constraints on: (a) power turbine speed, (b) power turbine inlet temperature, (c) power turbine torque, (d) propeller torque, and (e) propeller thrust. A digital simulation was employed to quantify the performance improvements available through coordinated control of engine power and propeller pitch during propulsion maneuvers. Finally, practical control implementation alternatives are considered.

The simulation based investigation has shown that significant maneuvering performance improvements are realizable using coordinated control of power and pitch. Reductions in stopping distance of about 20% relative to typical control techniques appear to be achievable.

2. INTRODUCTION

It has long been realized that the stopping potential of gas turbine powered ships with controllable pitch propellers is not being fully exploited by current programmed control systems. One previous investigation (1) considered optimal control of a steam/CPP propulsion system and demonstrated promising results may be achievable. It also indicated an optimal control algorithm was better suited to ships using high performance propulsion systems such as the one considered in this analysis. During conventional control algorithm design, the steady-state

characteristics for pitch and power are carefully considered while the transient characteristics typically allow the pitch to stroke at the maximum attainable rate limited by the pitch hydraulics while the power is applied at a rate designed to prevent damage to the machinery components. This design process is reflected in DD 963, CG 47, and DDG 993 class ships. While this concept can provide safe, predictable control characteristics, it does not produce the best attainable maneuvering characteristics.

The primary purpose of this paper is to quantify potential improvements in stopping performance using existing machinery systems. This potential is limited by the machinery characteristics which must apply the reversing propeller thrust (supplemented by the ship's resistance) to overcome the momentum of the ship's motion. To accomplish this, the concept of an idealized control system is introduced. The idealized control system coordinates engine power and propeller pitch to produce the fastest possible maneuvering without exceeding established machinery constraints. Performance is limited in this analysis by constraints imposed on: (a) power turbine speed, (b) power turbine inlet temperature, (c) power turbine torque, (d) propeller torque, and (e) propeller thrust. While this study was limited to these five specific constraints, the technique can be applied to more or different constraints. The study considers two maneuver types: crash back and crash forward each in both the one engine per shaft and two engines per shaft operating modes. The crash back is a maneuver initiated from the maximum ahead condition where maximum astern thrust is ordered to quickly stop the ship. The crash forward is the opposite maneuver from full astern. Using digital simulation it is shown that stopping distances in crash backs can be reduced by as much as 20% using idealized control compared to stopping distance using conventional control techniques. Improvements in crash forward performance by as much as 34% are also achievable. Nearly all of these performance improvements can be realized through the implementation of a feed forward controller which mimics the idealized engine power and propeller pitch control functions.

Section 3 describes the study ship analyzed herein and section 4 presents an overview of the mathematical models used in the simulation. Section 5 defines the constraints relative to the steady-state operating conditions. Sections 6 and 7 describe and compare the stopping scenario under idealized control, the practical alternative, and the typical scenario. Section 8

identifies the improvements attainable by reducing the specified propeller pitch stroke time. Section 9 summarizes the findings and presents conclusions.

3. DESCRIPTION OF THE STUDY SHIP

The study ship is a notional 8500 l.t. destroyer powered by two 18 foot diameter controllable pitch propellers. Each propeller is powered by one or two LM2500 marine gas turbines rated at about 25,000 hp. A Power Lever Angle (PLA) controller and a Main Fuel Controller typical of current LM2500 installations is included in this study ship. Figure 1 diagrams the basic control and machinery configuration for one of the two identical shafts.

The focus of this analysis concerns the variations in the outer loop control algorithms which drive the PLA controller and the propeller pitch ratio. Three types of outer loop control algorithms are considered. The first is referenced as "typical control" because it incorporates the basic characteristics of several (but not all) existing ship classes powered by LM2500 gas turbine engines. The second is referenced as "idealized control." This control algorithm coordinates control of propeller pitch and engine power to maximize the attainable reversing thrust. The last is referenced as "practical control," this control algorithm uses simple piece-wise linear engine and propeller commands to approximate the commands determined by the idealized control algorithm. Transients for each of these algorithms are shown in later sections.

The reversing performance of the ship is limited by various constraints (limits of safe operation) and the basic operating characteristics of the system. The constraints considered in this analysis are certain maximum limits set on:

(a) power turbine speed (equivalent to a propeller speed),
(b) propeller thrust (ahead and astern),
(c) propeller torque,
(d) power turbine torque, and
(e) power turbine inlet temperature.

Section 5 defines the specifics of these constraints. In addition, performance is limited by the basic operational limitations of the system including: maximum propeller pitch stroke rate, limits imposed by the main fuel controller, limits

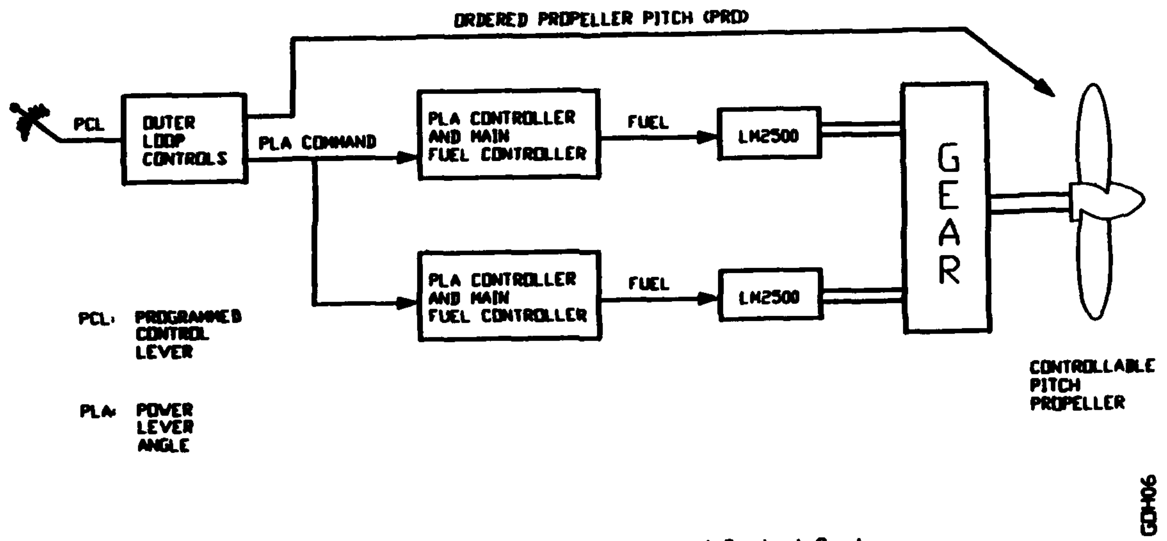


Figure 1. Simplified Study Ship Propulsion and Control System
(one of two identical shafts shown)

imposed by the PLA controller, maximum PLA, basic operating characteristics of the gas turbine, and hydrodynamic characteristics.

4. MATHEMATICAL MODELS

The analysis presented herein is based on detailed non-linear mathematical models of the ship system performance including the hull and propeller hydrodynamics, machinery, and control system. Although a detailed presentation of the models used for this analysis is beyond the scope of this paper, an overview is presented to aid the understanding of the analysis and techniques involved. Most of these models are presented in previous Ship Control Systems Symposia as well as other publications, the details are not repeated here (1, 2, 3, 4). Substantial details are presented describing the idealized control system algorithm which is new and provides the basis of this paper.

4.1 Hydrodynamic Models

Mathematical models related to the hydrodynamics include thrust deduction factor, wake factor, ship's mass including entrained water, ship resistance, propeller torque, and propeller thrust. The ship mass including entrained water is taken as 8500 l.t. plus 8% for entrained water. The other hydrodynamic models are based on a review of existing hydrodynamic test data for various destroyers and cruisers; models were then developed which are typical of an 8500 l.t. destroyer. Thus, to avoid security classification the study was not done for any specific ship but rather for a notional 8500 l.t. destroyer. Repeating this analysis for a specific ship class would involve replacing existing notional data with ship specific data.

4.2 Machinery Models

The primary machinery system models include the propeller pitch stroke rate, rotational inertia of the drive train (power turbine, reduction gear, shafting, propeller and entrained water), LM2500 gas turbine, PLA controller and main fuel controller. The propeller pitch stroke rate is based on a 24 second pitch stroke time from full ahead to full astern. Field testing indicates this is readily achievable although longer times are also seen. For analysis involving the typical control algorithm, both a 30 second and 24 second pitch stroke time is

considered since typical specifications allow pitch stroke rates of 30 seconds. The rotational inertia of the drive train is typical of existing ships and ships currently under construction.

The mathematical models of the PLA controller, main fuel controller and LM2500 gas turbine are based on the designs aboard current US Navy ships. The PLA controller model reflects existing electronics including the PLA rate limiter, torque limiter, topping governor, and acceleration limiter. The main fuel controller reflects typical engine mounted systems including the main speed schedule, acceleration and deceleration limiters, and the jump and rate limiters. The LM2500 model is a thermodynamic model including representations of flows, temperatures and pressures throughout the engine. The primary LM2500 components modeled include the combustor, the compressor, the high pressure turbine and the power turbine. The models of the PLA controller, main fuel controller and the LM2500 have been successfully validated against at sea test data based on several ship trials.

4.3 Control System Models

The control system models discussed here refer to the outer loop controls identified in Figure 1. The models are the representation of the control algorithms which issue commands to the propeller pitch system and the PLA controller, this is typically part of a programmed control system on modern gas turbine ships. These algorithms can be implemented in a variety of ways, most likely using a digital computer. Typical control, idealized control and practical control algorithms are discussed below.

a. Typical control. Since this paper presents attainable improvements in stopping performance, it was necessary to define a baseline system. The baseline system selected is similar to the CG 47/DDG 993/DD 963 classes which have many characteristics common with the gas turbine high speed mode of the FGG 511 and FCG 612 class ships. This algorithm was selected as the typical control algorithm because of its common usage and because it is considered to achieve rapid stopping performance. The essential characteristics of this control algorithm during a rapid reversal are:

(a) Stroke pitch from existing condition to full astern (or to full ahead for a crash forward). Stroke is limited by the pitch hydraulics.

(b) Cut power to idle until the propeller pitch nearly reverses.

(c) Reapply power but do not let PLA exceed preset limits based on measured propeller pitch.

(d) Apply power as necessary to achieve the final ordered propeller speed unless the torque limit is exceeded. The typical control algorithm limits power based on measured propeller shaft torque while the PLA controller also limits power application based on calculated power turbine torque. In the reversals considered in this analysis, the torque limiters governed the final portions of these maneuvers.

The typical control algorithm has no direct means to limit power turbine inlet temperature or propeller thrust; thus, when comparisons are made between typical control and idealized control responses, the typical control often allows these parameters to exceed the maximum values allowed by the idealized control.

b. Idealized control. The idealized control algorithm was developed to continuously coordinate gas turbine power command (PLA) and propeller pitch ratio (PR), such that the magnitude of developed thrust is maximized subject to the system constraints and subject to the dynamics of the system. The algorithm considers the current state of the system (e.g., ship speed, propeller speed, propeller pitch, and gas generator speed), and varies the control signals, PLA and ordered pitch ratio (PRO), to produce the maximum (in magnitude) thrust one second into the future. This is done subject to the predefined system constraints. Constraints are imposed on power turbine speed (NPT), power turbine inlet temperature (T54), power turbine torque (QPT), propeller torque (Q), and propeller thrust (T).

The means of determining the control signals is briefly described here and in Figure 2. More details are provided later in this section. The algorithm first determines a "region of realizable control." This is an approximation to the set of future system states realizable in one second from the current time. This region is represented in the propeller speed-propeller pitch ratio plane since lines of constant propeller torque and thrust are easily represented in this plane. This

Establish Region of Realizable Control (RRC)

Limit RRC to prevent overtorque, overthrust, and overspeed (allowable RRC)

Select point of maximum/minimum thrust within allowable RRC

Translate selected point to ordered gas generator speed and ordered pitch ratio

Reduce ordered gas generator speed and correct ordered pitch ratio if needed to prevent excessive power turbine inlet temperature

Filter ordered gas generator speed

Convert to PLA command

Figure 2. Steps in Determining Control Signals, Power Level Angle and Ordered Pitch Ratio

region is further restricted by the constraints on propeller speed, propeller thrust, propeller torque, and engine torque, thereby yielding an allowable region of realizable control. Thrust is evaluated within the allowable region to select the point with the maximum (in magnitude) thrust. This point represents the desired state to be realized one second into the future.

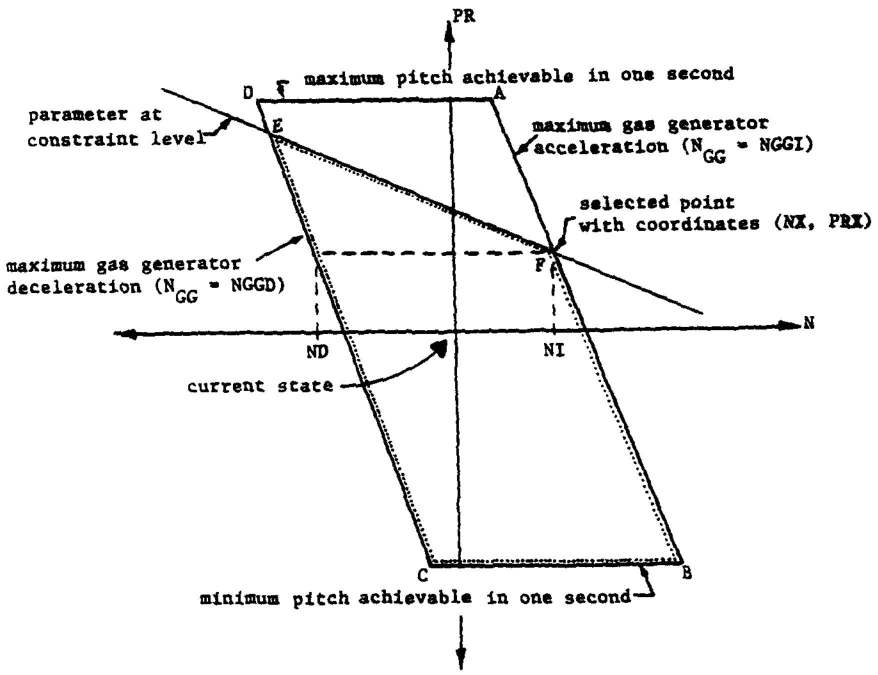
The selected point has coordinates which are translated into an ordered pitch ratio and ordered gas generator speed (NGGDMO). The constraint on power turbine inlet temperature (T54) is implemented using an integral type feedback controller, which modifies NGGDMO accordingly. Finally, a first order lag filter with a 1.0 second time constant is employed to smooth the signal. Ultimately, the adjusted NGG demand signal is translated into an ordered PLA. This algorithm is executed periodically at a rate of 10 Hz.

The method of determining the control signals is described in greater detail immediately below and is functionally presented in Figures 2 and 3.

The first step of the algorithm is to mathematically define the "region of realizable control." This region corresponds to an approximation of the set of all possible future system states realizable after one second from the current time. This region is graphically constructed in the propeller speed, propeller pitch ratio plane (N-PR plane) as shown in Figure 3. The procedure for this construction involves the placement of four straight lines around the current system state which approximate:

(a) maximum pitch achievable in one second,
(b) minimum pitch achievable in one second,
(c) system states with maximum gas generator acceleration in one second,
(d) system states with maximum gas generator deceleration in one second.

The values of maximum and minimum pitch achievable in one second are readily determined from the pitch stroke rate with the added provision that pitch may not exceed its upper and lower mechanical bounds. The placement of the gas generator acceleration/deceleration limit lines is accomplished with the aid of a simplified gas generator model. Briefly, forward



- Region of Realizable Control is the set of future states realizable in one second. Outlined by the solid line (region A B C D).
- Allowable Region of Realizable Control is the set of future states realizable in one second without exceeding constraints. Outlined by the dotted line (region B C E F).
- Conversion of Selected Point to ordered Gas Generator Speed; NGGDMO = NGGD + (NX - ND)/(NI - ND) * (NGGI - NGGD)

Figure 3. Region of Realizable Control

integrations for one second are carried out to determine the propeller speed (and other parameters) which result with:

(A) maximum gas generator acceleration and maximum rate of pitch increase,
(B) maximum gas generator acceleration and maximum rate of pitch decrease,
(C) maximum gas generator deceleration and maximum rate of pitch decrease, and
(D) maximum gas generator deceleration and maximum rate of pitch increase.

Knowledge of these four propeller speeds enables the placement of four points (A, B, C, D) in the N-PR plane which are linearly connected as shown in Figure 3 to enclose a region. This region shall hereafter be referred to as the region of realizable control (RRC).

Having established the RRC in the N-PR plane, consideration is now given to excluding that portion of the region which would produce violations of the predefined system constraints. The portion of the RRC which is not excluded after the thrust, propeller torque, and engine torque constraints have been considered is called the "allowable region of realizable control."

The forward integration produces a value of propeller thrust for each of the four control cases, thus the thrust associated with each corner of Figure 3 is known. A test is made to determine whether any of these values violates the constraint value. If no violation is found, the algorithm proceeds to the torque constraint. If a violation is found, a line representing thrust equal to the thrust constraint is added to the N-PR plane as shown in Figure 3 as the "parameter at constraint level" line. While lines of constant thrust are not straight lines in the N-PR plane, they may be reasonably approximated as such for small variations in N and PR. The position of this line is established by determining the coordinates of two points on it, one on each of the gas generator response limit lines. In a crash forward, the thrust constraint is positive, and all points above this line are excluded from the allowable RRC. In a crash back, the thrust constraint is negative, and all points below this line are excluded.

The exclusion of the portion of the RRC which would produce violations of the propeller torque constraint is very similar to the procedure described above for the thrust constraint. The differences are all an outgrowth of the fact that propeller torque does not increase monotonically with increasing pitch as is the case with thrust. A check is also made to determine whether the current value of propeller torque violates the propeller torque constraint.

The exclusion of the portion of the RRC which would produce violations of the engine torque constraint is fully analogous to the procedure used for the propeller torque constraint, with the addition that the effects of shaft acceleration are added when the constraint is translated from the propeller to the engine.

The portion of the RRC which lies to the right of a vertical line corresponding to the power turbine speed constraint is excluded to prevent overspeed of the power turbine. This is equivalent to limiting the maximum propeller speed which differs from the power turbine speed by the gear ratio of 22.5.

At this point, the RRC has been reduced to an allowable RRC, and a desired future state is to be selected from the points therein. The criteria for selection is the greatest thrust for crash forwards and the most negative thrust for crash backs. It is assumed that the extreme of thrust will be found on the boundary of the allowable RRC. Therefore, the algorithm evaluates the thrust at the vertices which form the allowable region of the realizable control, as indicated in Figure 3, and selects the point of greatest thrust magnitude. It is possible that adjacent vertices may have the same thrust, in which case the point which has the greater propeller speed is chosen. This is done because torque loads are usually reduced as propeller speed is increased at constant thrust, thereby increasing the margin between actual propeller torque and the propeller torque constraint. The selected point has coordinates of propeller speed and pitch ratio called NX and PRX, respectively (shown on Figure 3).

Ordered pitch ratio (PRO) is found in a straightforward manner by ordering pitch to change at a rate which would cause a change from the current value (PR) to the selected value (PRX) in one second.

An integral type feedback controller is used to modify the NGGDMO signal to limit T54 to the constraint value. When this constraint causes limiting of gas generator speed demand, the ordered pitch ratio is compensated to maintain the selected propeller speed, NX. In such cases the developed thrust may be slightly less than the optimum achievable.

Finally, a first order lag filter with the time constant of 1.0 seconds is used to smooth the demand for gas generator speed. Ultimately, the filtered NGG demand is converted to a PLA command based on the known operational characteristics of the MFC.

__c. Practical control.__ The practical alternative control algorithm is included in this paper to help estimate the sensitivity of ship performance to variations in control commands relative to the idealized control algorithm. Thus, it is included to approximate variations which may be expected when the principles of the idealized control system are implemented shipboard.

The practical control algorithm was implemented as a series of piece-wise linear pitch and PLA commands to mimic the results of an idealized control system maneuver. In general, these transients approximate the pitch commands found with the idealized control system while the PLA commands were kept slightly below the idealized control system value. Thus, the practical control is somewhat conservative.

5. STEADY-STATE OPERATING CONDITIONS AND SELECTION OF CON[illegible]RAINT VALUES

Eac[illegible]a[illegible]uvers considered in this paper begins from a steady-stat[illegible]dition at either maximum ahead or maximum astern in either the one engine per shaft or two engines per shaft mode. These steady-state conditions are defined in Table 1.

The selection of specific constraint values is critical to the design of a control algorithm and is dependent on the specific ship application. The constraints selected for this notional ship are as follows:

Power Turbine

Speed Constraint	3672 rpm
Inlet Temperature Constraint	1530°F
Torque Constraint	48,300 ft-lbf

Propeller

Torque	1,738,000 ft-lbf
Thrust (ahead and astern)	448,900 lbf

The torque constraints are 8% above the maximum steady-state condition and the thrust constraint is 25% above the maximum steady-state condition. The power turbine speed and inlet temperature constraints were selected based on typical LM2500 applications. These constraints are the same for ahead and astern conditions.

All maneuvers considered herein were analyzed at a compressor inlet temperature of 100°F.

Table 1. Summary of Steady-State Conditions

Parameter	Maximum Ahead		Maximum Astern	
	One Engine	Two Engines	One Engine	Two Engines
Propeller				
Speed [rpm]	132.0	160.0	115.0	115.0
Pitch [%]	100.0	100.0	-48.7	-48.7
Torque [ft-lbf]	$983 \cdot 10^3$	$1609 \cdot 10^3$	$809 \cdot 10^3$	$809 \cdot 10^3$
Thrust [lbf]	210,800	359,100	-114,200	-114,200
Power Turbine				
Torque [ft-lbf]	44,740	36,420	36,870	18,420
Speed [rpm]	2972	3600	2588	2588
Inlet Temp [°F]	1454	1461	1313	1076
General				
PLA [deg]	107.1	102.7	92.9	72.7
Ship Speed [knots]	28.60	32.96	-17.35	-17.35

6. STOPPING SCENARIO UNDER IDEALIZED CONTROL

A detailed review of the one engine per shaft crash back using idealized control is contained in the text and Figure 4. Many of the same characteristics are seen for the other maneuvers which are not discussed in detail.

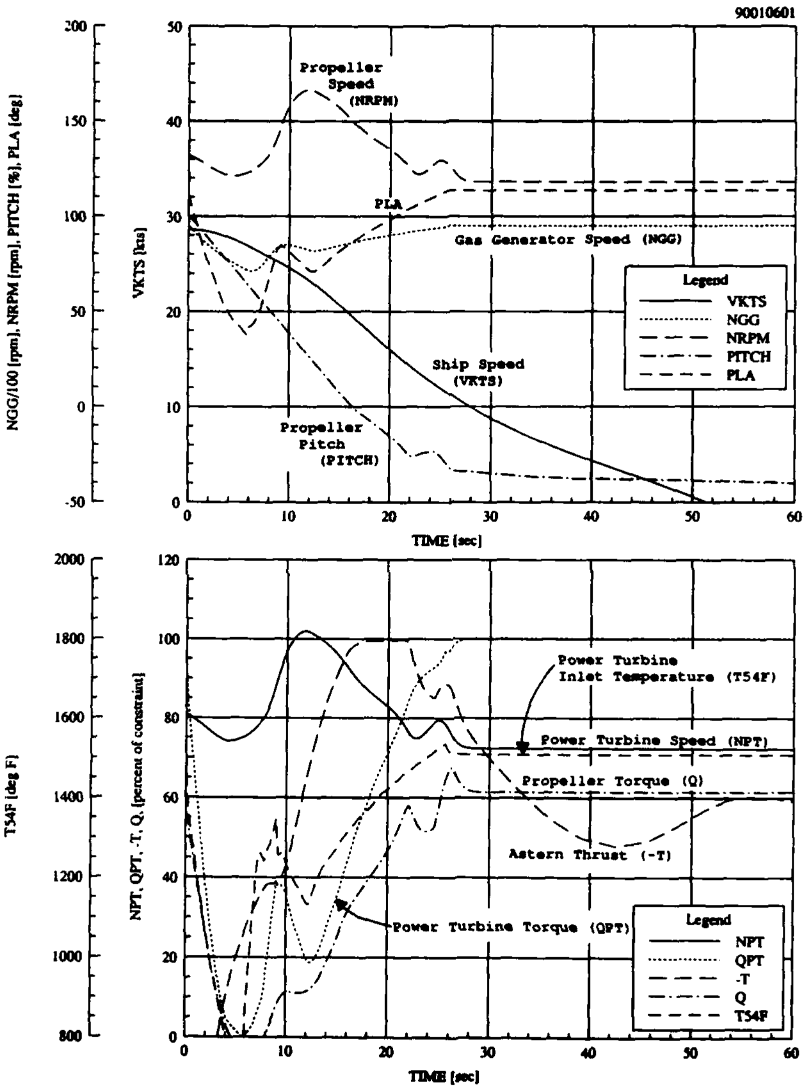


Figure 4. Idealized Control
One Engine per Shaft Crash Back

The parameters shown in figure 4 are:

VKTS	Ship speed [knots]
NGG	Gas generator speed [rpm]
NRPM	Propeller speed [rpm]
PITCH	Propeller pitch [%]
PLA	Gas turbine power lever angle [deg]
NPT	Power turbine speed [% of constraint value]
QPT	Power turbine torque [% of constraint value]
T54F	Power turbine inlet temperature [°F]
T	Propeller thrust [% of constraint value]
Q	Propeller torque [% of constraint value]

Four of the constrained parameters are presented as a percentage of the appropriate constraint value. For example, QPT=100% indicates the power turbine torque is at its constraint value of 48,300 ft-lbf. This highlights which constrained parameter is the limiting factor at any given time. Note that in order to achieve the maximum attainable reversing thrust throughout the maneuver, each of the various constraints may become the limiting factor at various times during the maneuver. For ease in presentation, the astern thrust (-T) is shown for crash back maneuvers while ahead thrust (T) is shown for crash forward maneuvers.

All idealized control reversals can best be understood by dividing the maneuver into several different regions. For the one engine per shaft crash back, the maneuver can be divided into six independent regions.

(1) At the initial condition the ship is operating at 28.6 knots ahead, 100% propeller pitch, and 132 rpm propeller speed. The first region (0 to 6 seconds) is characterized by rapid power and pitch reductions. PLA is controlled to yield the fastest possible reduction in fuel flow rate as limited by the MFC. Propeller pitch is reduced at the maximum rate allowed by the pitch hydraulics.

(2) The second region (6 to 16 seconds) begins after a reverse thrust is achieved and is characterized by rapid pitch reduction and reapplication of power. The reapplication of power is limited by the power turbine speed constraint.

(3) The third region (16 to 24 seconds) is marked by changes in the behavior of both PLA and pitch control. In this region the reduction in pitch is slower and the reapplication of PLA is limited by the propeller thrust constraint.

(4) The fourth region (24 to 25 seconds) represents a period where PLA is briefly limited to prevent the power turbine inlet temperature from exceeding its constraint of 1530°F.

(5) The final transient region (25 seconds and beyond) is characterized by PLA at it's limit of 113.5 degrees and propeller pitch being controlled to maximize reversing thrust while not exceeding the power turbine torque constraint. Additional reductions in pitch (toward full astern) would yield additional reversing thrust but would cause the power turbine torque to exceed its constraint.

(6) When the final astern operating condition is reached, the propeller pitch and PLA will be reduced -48.7% and 92.9 degrees, respectively.

In summary, this and other idealized control reversals share the following basic characteristics: (a) initial rapid reversal of pitch and reduction of power, (b) gradual reapplication of power soon after thrust direction reverses, (c) very slow continuation of pitch reversal for the last phases of pitch stroke, and (d) continuation of power increase limited by the constrained parameters.

7. COMPARISON OF STOPPING SCENARIO UNDER IDEALIZED, PRACTICAL, AND TYPICAL CONTROL ALGORITHMS

For the reversals considered, each was simulated with idealized control, practical control, and typical control. Each of those maneuvers is presented in this section. Typical control presented in this section assumes a 30 second pitch stroke time, section 8 addresses the effect of a 24 second pitch stroke time.

7.1 One engine per shaft crash back

Idealized, practical, and typical control are compared for a one engine per shaft crash back in Figure 5. Practical control

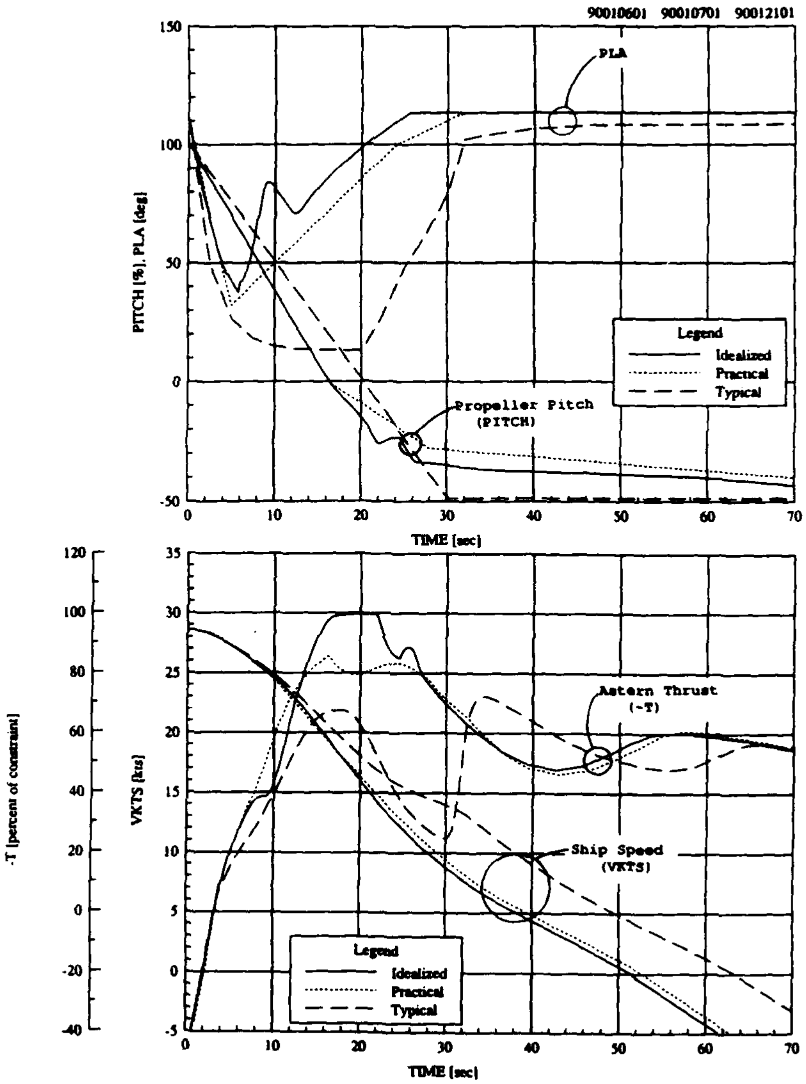


Figure 5. Idealized, Practical, and Typical Control
One Engine per Shaft Crash Back

is intended to approximate idealized control but demonstrate sensitivity in stopping performance to small deviations from idealized control. Although the differences in commands are significant for much of the maneuver, there is little effect on overall system performance. Thus, it is seen that the concepts found with the idealized control system need not be implemented exactly to realize the available stopping improvements.

Direct comparison of idealized control with typical control reveals no significant difference exists for the first 10 seconds. However, typical control delays reapplication of power until pitch has reversed while idealized control reapplies power much sooner. The additional delay in power reapplication is not needed because the ahead ship speed shifts the point of minimum torque load to be well above zero pitch. The difference between idealized and typical becomes greater when the typical control pitch reaches full astern; this is because the full astern pitch yields higher torque values which force a lower PLA and therefore reduces reversing thrust. Recall that the idealized control system controlled pitch to limit torque (rather than controlling PLA to limit torque) thereby allowing higher speeds and higher power application resulting in greater reversing thrust.

Maneuvering performance is summarized in Table 2. Briefly, the idealized control offers a reduction in stopping distance of 20.4% compared to typical control. Practical control offers an 18.5% reduction.

7.2 Two engine per shaft crash back

Idealized, practical, and typical control are compared for a two engine crash back in Figure 6. As with the one engine per shaft crash back, the practical control algorithm yielded essentially the same performance as the idealized control algorithm. Thus, it is again seen that the concepts found with the idealized control system need not be implemented exactly to realize the available stopping improvements.

Direct comparison of idealized control with typical control indicates the same general characteristics observed with the one engine per shaft crash back hold for the two engine per shaft crash back. However, for the one engine typical control case the limiting factor was power turbine torque, while propeller torque is the limiting factor for the two engine per shaft case. In addition, the typical control did not successfully limit thrust

Table 2. Summary of Performance

RUN	MANEUVER (1)	ENGINES PER SHAFT	CONTROL ALGORITHM (2)	STOPPING TIME [sec]	STOPPING DISTANCE		PEAK THRUST [% of constraint]
					DISTANCE [ft]	CHANGE (3) [%]	
90010601	CB	1	Idealized	51.4	1158	-20.4	- 99.8
90010701	CB	1	Practical	52.8	1186	-18.5	- 85.5
90012101	CB	1	Typical, 30 sec	63.3	1455	---	- 72.3
90010801	CB	1	Typical, 24 sec	59.6	1336	- 8.2	- 78.4
90041801	CB	1	Practical (4)	54.4	1207	-17.0	- 82.5
90010606	CB	2	Idealized	45.1	1228	-16.2	-100.5
90012802	CB	2	Practical	46.3	1246	-14.9	- 98.8
90012102	CB	2	Typical, 30 sec	52.9	1465	---	-105.8
90010802	CB	2	Typical, 24 sec	48.6	1323	- 9.7	-113.3
90010603	CF	1	Idealized	29.4	530	-34.2	101.9
90010703	CF	1	Practical	31.3	564	-29.9	99.5
90012103	CF	1	Typical, 30 sec	47.5	805	---	63.6
90010803	CF	1	Typical, 24 sec	43.5	731	- 9.2	65.2
90010604	CF	2	Idealized	29.2	527	-27.0	100.3
90010704	CF	2	Practical	32.3	574	-20.5	97.3
90012104	CF	2	Typical, 30 sec	39.3	722	---	103.9
90010804	CF	2	Typical, 24 sec	35.3	645	-10.7	104.0

Notes:

(1) CB = Crash Back, CF = Crash Forward.
(2) Algorithms defined in Section 4, the times (30 and 24 seconds) refer to the pitch stroke time.
(3) Change in stopping distance is relative to typical control with 30 second pitch stroke time. Negative change represents reduced stopping distance.
(4) Transient PLA reduced 5 degrees below 90010701.

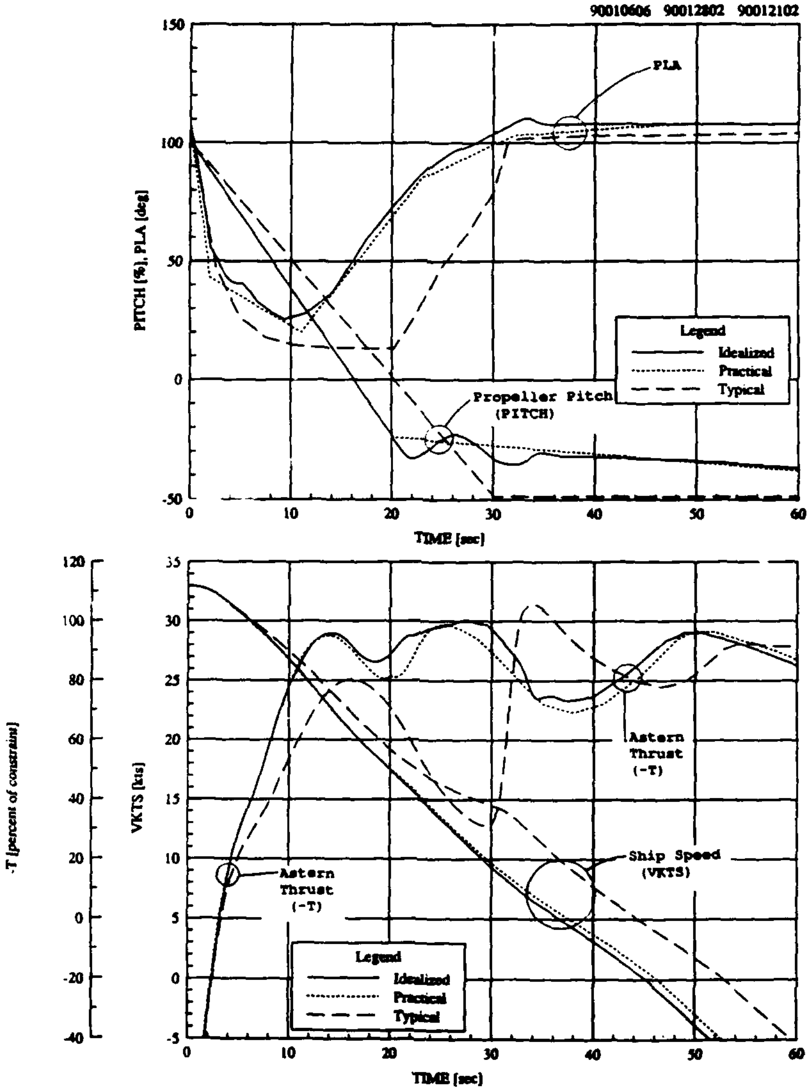


Figure 6. Idealized, Practical, and Typical Control
Two Engines per Shaft Crash Back

to be within the allowable thrust constraint; thus, typical control actually violated the limits imposed on the idealized control case. If the idealized control system were allowed higher thrust loads, some small additional performance improvements could be expected.

Table 2 indicates the idealized control offers a reduction in stopping distance of 16.2% compared to typical control. Practical control offers a 14.9% reduction.

7.3 One engine per shaft crash forward

Idealized, practical, and typical control are compared for a one engine per shaft crash forward in Figure 7. As with the crash back maneuvers, the idealized and practical control algorithms yield essentially the same system performance.

Direct comparison of idealized control with typical control reveals similar performance only for the first 11 seconds, thereafter, the idealized control system demonstrates much faster reversing characteristics without exceeding the predefined constraints. There are two primary reasons for this difference. First, the typical control system does not reapply power as soon as possible, and when it is reapplied, it is applied at a slow rate corresponding to the gradual increase in propeller pitch. Second, typical control increases pitch to full ahead relatively quickly thereby causing torque loads to be at their limit without attaining the high levels of thrust achievable with less propeller pitch. These characteristics are also seen in the other maneuvers.

Table 2 indicates the idealized control offers a reduction in stopping distance of 34.2% compared to typical control. Practical control offers a 29.9% reduction.

7.4 Two engine per shaft crash forward

Idealized, practical, and typical control is compared for a two engine per shaft crash forward in Figure 8. As with the one engine per shaft crash forward, the practical control algorithm yielded essentially the same performance as the idealized control algorithm.

Direct comparison of idealized control with typical control indicates the same general characteristics observed with the one

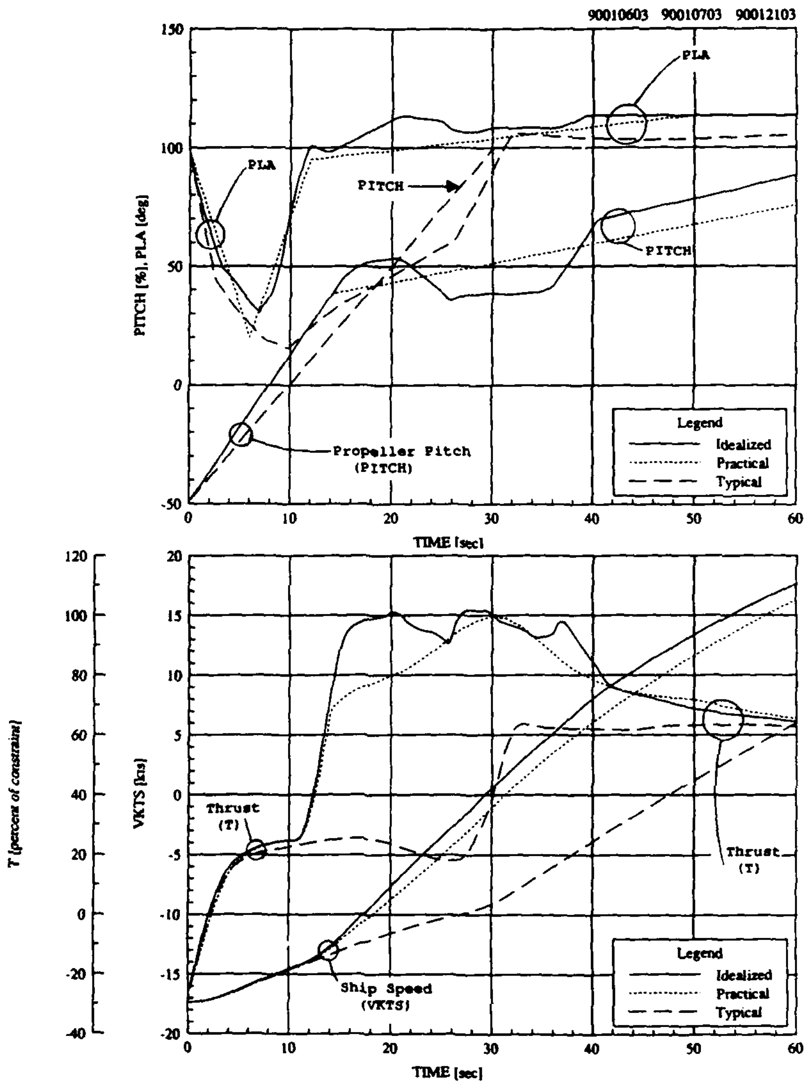


Figure 7. Idealized, Practical, and Typical Control
One Engine per Shaft Crash Forward

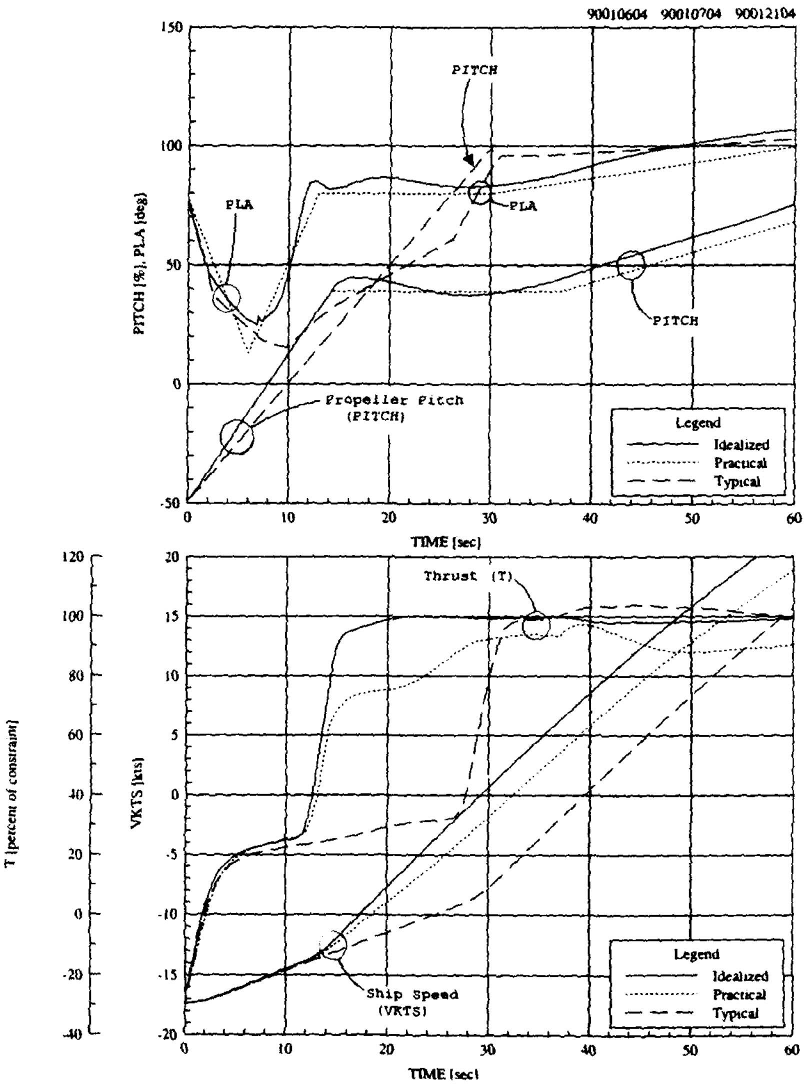


Figure 8. Idealized, Practical, and Typical Control
Two Engines per Shaft Crash Forward

engine per shaft crash forward hold for the two engine crash forward.

Table 2 indicates the idealized control offers a reduction in stopping distance of 27.0% compared to typical control. Practical control offers a 20.5% reduction.

8. ATTAINABLE IMPROVEMENT IN TYPICAL CONTROL ALGORITHM BY DECREASING PROPELLER PITCH STROKE TIME

One reason the typical control algorithm yields slower stopping performance is the slower pitch stroke rate. Although the idealized algorithm indicates a slow pitch stroke rate is desirable for the final portions of a reversal, a faster rate can be used to quickly reverse the direction of thrust during the first few seconds of a reversal. The initial portion of a reversal is critical because the relatively high ship speed translates into substantial distance traveled.

To quantify this difference, additional maneuvers were simulated using the typical control algorithm with a 24 second pitch stroke time. Figure 9 compares typical control with both 24 and 30 second pitch stroke times to idealized control for the two engine per shaft crash back. Although a 9.7% improvement in stopping distance can be realized using the faster pitch stroke rate with typical control, the peak thrust increases to be 13.3% above the constraint value. This is compared to a 16.2% improvement in stopping distance realized through idealized control which does not suffer this increase in peak thrust. Table 2 summarizes the peak thrust values and stopping distances for idealized control and typical control with 24 and 30 second pitch stroke times.

While faster pitch stroke rates yield substantial reductions in stopping distance, the increase in peak thrust load is significant. Therefore, it may not be possible to implement faster pitch stroke rates without corresponding improvements in the control algorithm to limit peak thrust. Alternatively, if the thrust bearings can withstand the increased thrust loads, the faster pitch stroke rate may be beneficial.

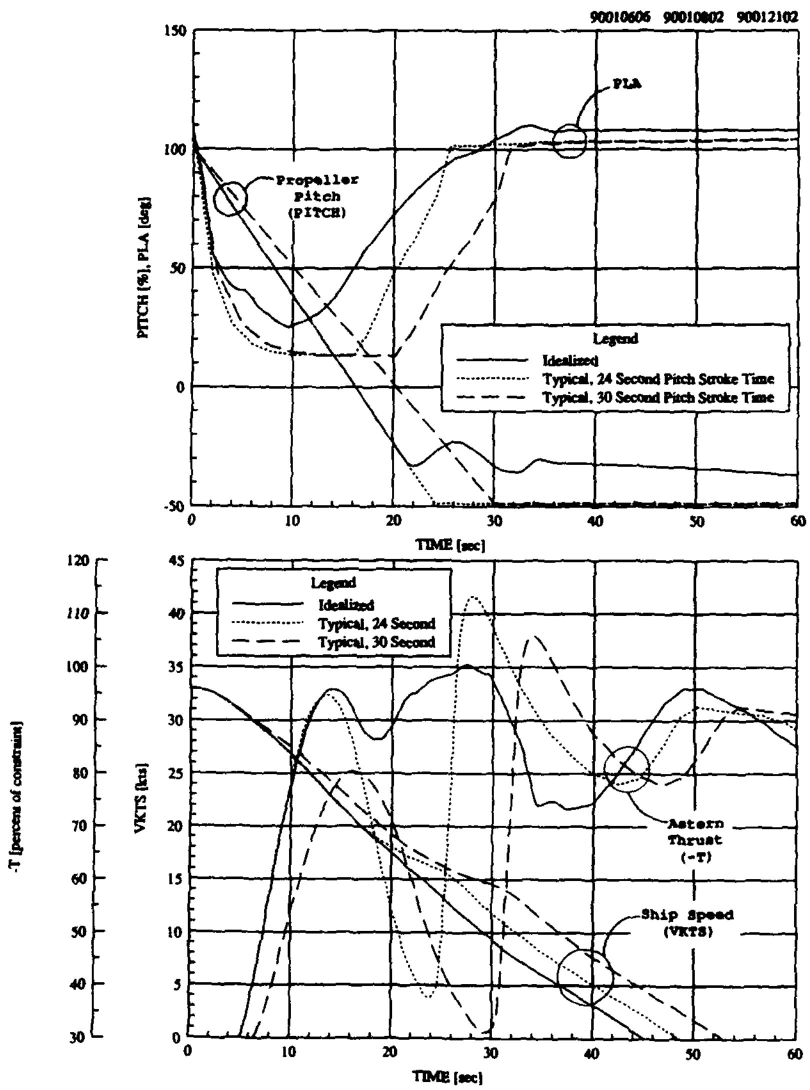


Figure 9. Idealized and Typical Control
Two Engines per Shaft Crash Back

9. SUMMARY AND CONCLUSIONS

(a) The concept of the idealized control system provides a useful tool for control system designers to improve stopping performance without increasing peak machinery loads. By simulating the idealized control system actions the designer can determine the desired transient characteristics of power and propeller pitch commands. Then, the control algorithm can be designed with the knowledge of the idealized control system response. Although this paper addresses a gas turbine controllable pitch propeller machinery system, the technique can be applied to other types of machinery systems.

(b) Potential reductions in stopping distance relative to typical control systems in use today are significant.

--- 20.4% reduction for one engine per shaft crash back.

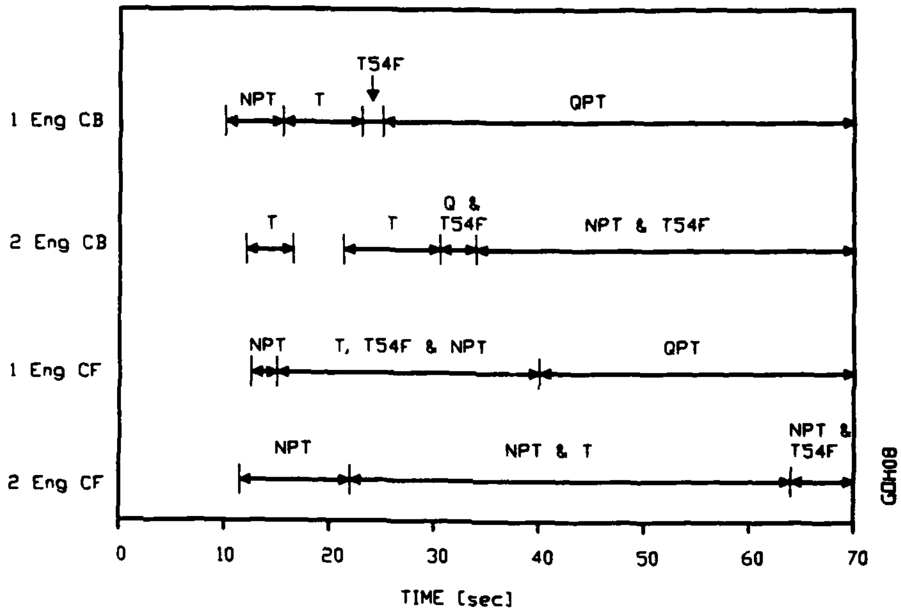
--- 16.2% reduction for two engines per shaft crash back.

--- 34.2% reduction for one engine per shaft crash forward.

--- 27.0% reduction for two engines per shaft crash forward.

These reductions can be realized through modifications to the control algorithm without increasing peak machinery loads. However, these algorithms must be developed to control the system loads to correspond to different system constraints during each portion of the maneuver as indicated in figure 10.

(c) Although the idealized control algorithm provides a specific transient for ordered PLA and ordered propeller pitch, the sensitivity of ship performance to variations in these commands is relatively small. Thus, it is not necessary to exactly reproduce the idealized control commands to realize nearly all of the available improvement in stopping distance. For example, for each degree PLA is below its idealized



NOTES: T - Propeller thrust constraint active
Q - Propeller torque constraint active
QPT - Power turbine torque constraint active
NPT - Power turbine speed constraint active
T54F - Power turbine inlet temperature constraint active

Figure 10. Idealized Control Active Constraint Summary

level, a degradation of only four feet in stopping distance is seen for the one engine per shaft crash back.

(d) The idealized control system algorithm indicates the propeller pitch should be reduced as quickly as possible for the initial phases of a reversal. However, after about two-thirds of the pitch stroke, additional pitch stroke should take place very slowly and should be controlled in a manner to yield a high rotational speed while operating at or near the torque constraint. The exact characteristics are dependent on the specific ship application.

(e) Although not presented herein, the idealized control system concept has been applied to acceleration maneuvers; similar improvement potential was found.

(f) The stopping distance observed with typical control can be reduced significantly by simply reducing the pitch stroke time. However, peak propeller thrust loads show a significant increase which may exceed thrust bearing limits.

REFERENCES

(1) J.W. Donnelly "Propulsion Control Optimization of CRP Propeller Driven Ships," Proceedings of the Fifth Ship Control Systems Symposium, October 30 - November 3, 1978. Volume 5, p. Q2, 3-1.

(2) C.J. Rubis and T.R. Harper "Reversing Dynamics of a Gas Turbine Ship with Controllable-Pitch Propeller," Proceedings of the Fifth Ship Control Systems Symposium, October 30 - November 3, 1978. Volume 2, p. E2, 1-1.

(3) C.J. Rubis "Acceleration and Steady-State Propulsion Dynamics of a Gas Turbine Ship with Controllable-Pitch Propellers," SNAME Transactions, Vol 80, 1972.

(4) T.R. Harper and L.C. Carroll "The Effect of Controllable-Pitch Propeller Data Interpolation Procedures on Propulsion Plant Simulations," Summer Computer Simulation Conference, 1977.

EXPERIENCE WITH CONTROLLABLE PITCH PROPELLERS DURING FULL SCALE PERFORMANCE AND SPECIAL TRIALS

Richard J. Stenson
Michael L. Klitsch
and Everett L. Woo
Full Scale Trials Branch
David Taylor Research Center

1. ABSTRACT

This paper presents analysis problems experienced with controllable pitch propellers during Performance and Special Trials conducted for Naval Sea Systems Command (NAVSEA) by the David Taylor Research Center (DTRC). The effects of propeller pitch on ship's powering is documented with full scale trials data. Also included is data showing that ships being delivered to the U.S. Navy with controllable pitch propellers are not developing full power at, what the feedback control system indicates, is design propeller pitch. The controllable pitch propeller system is briefly described. The method currently used by DTRC to measure and calibrate propeller pitch prior to a sea trial is presented. Next, a discussion of how the propeller pitch changes without the feedback control system changing is presented. Finally, the paper closes with a recommendation for incorporating accurate in–hub pitch sensors in future U.S. Navy ships.

2. INTRODUCTION

This paper presents an examination of the controllable pitch propeller system currently installed on many U.S. Navy ships and its effect on ship's powering. Propeller pitch is not accurately portrayed due to inherent problems in currently used controllable pitch propeller systems. Propeller pitch can vary, without the feedback control system sensing any change due to temperature variations in the propeller system and shaft compression due to propeller shaft thrust. Therefore, propeller pitch is inaccurately indicated.

David Taylor Research Center (DTRC) conducts Performance and Special Trials for Naval Sea Systems Command (NAVSEA) on the lead ship of each class of ship built for the U.S. Navy. Propeller pitch has a very significant effect on ship's propeller rpm and propeller shaft torque and a smaller effect on ship's power and thrust. These factors are interrelated. At maximum design horsepower, the shaft rpm, shaft torque and propeller pitch should attain or be very close to design. This has not proved to be the case. In order to attain design horsepower the propeller(s) have to be operated at a greater than design pitch. This would seem to indicate a hydrodynamic problem such as a mismatch of the propellers and hulls.

DTRC also participates in Builders and Acceptance Trials (BT and AT) on most new ships built for the Navy, supplying personnel, torsionmeters, and rpm counters as Government Furnished Equipment (GFE) to determine shaft torque, shaft rpm, and shaft horsepower during the required full power demonstration. It has been our experience over recent years that the majority of the ships being delivered to the U.S. Navy with controllable pitch propellers are not developing full power at design propeller pitch as indicated by the feedback control system.

On some of the systems, the propeller rpm is programmed to be reduced during maneuvers to prevent overtorquing of the propeller shaft. This results in nonstandard turning maneuvers and impacts the analysis and comparison of turning circle parameters.

The paper begins with a discussion of the effects of propeller pitch on ship's powering. This provides data showing that propeller shaft torque and shaft speed will move in opposite directions from each other (as one increases the other will decrease) as propeller pitch changes. This section of the paper will also present a discussion of ships which were not able to achieve design full power at design pitch, as well as the effect of propeller pitch on tactical circles. The paper then provides a description of the controllable pitch propeller system, currently installed on U.S. Navy ships, on which DTRC has conducted trials. The method currently used by DTRC to measure and calibrate propeller pitch prior to a trial is then discussed. This method has been developed in recent years to help account for the inherent temperature and thrust problems in determining the propeller pitch. Next, a discussion is presented on the insen–sitivity of the feedback control system to changes in propeller pitch due to temperature variations in the propeller shaft system and propeller shaft compression due to thrust. Finally, the paper will close with a recommendation for incorporating accurate in–hub pitch sensors in future U.S. Navy ships.

3. EFFECT OF PROPELLER PITCH ON SHIP POWERING CHARACTERISTICS

The effects of propeller pitch on a ship's powering characteristics, the inability to make design full power at design pitch, and the effect of propeller pitch on tactical circle trials are discussed in the following section of the paper. The first part of the section discusses the effect that propeller pitch has on ship powering characteristics as portrayed by data from seven ship trials. A discussion of the ships' inability to develop design full power at design propeller pitch will follow. Finally, the effect of the controllable pitch propeller system on conducting tactical circle trials is presented.

3.1. Propeller Pitch Effects on Ship Powering Characteristics

Propeller pitch has a significant effect on the powering characteristics of ships. This is evidenced in the family of curves presented in Figures 1 through 7, and the data tabulated in Tables 1 and 2. Each of the figures presents propeller shaft rpm, propeller shaft torque, and shaft horsepower as a function of ship speed for three separate nominal propeller pitch conditions: Under Design, Design, and Over Design. Two of these figures, Figures 1 and 4 also include propeller thrust curves. All of the data presented in these figures were collected during Naval Sea Systems Command Performance and Special Trials conducted by the Center. These trials are documented in references 1 through 6. The trials were conducted on a number of different ships, FFG 7, FFG 36, LSD 41, CG 49, ARS 50, MCM 1, and T-AO 193. They include single screw gas turbine propulsion, twin screw gas turbine propulsion, and twin screw diesel propulsion. The ships range in size from the MCM 1 displacing 1,290 tons to the T-AO 193 displacing 39,700 tons.

As can be seen in these figures, increasing propeller pitch beyond Design (100%) pitch results in an increase in propeller shaft torque, a decrease in propeller shaft rpm, an increase in propeller thrust (on one trial), and essentially no change in shaft horsepower for a constant ship speed. Decreasing propeller pitch below Design (100%) pitch results in a decrease in propeller

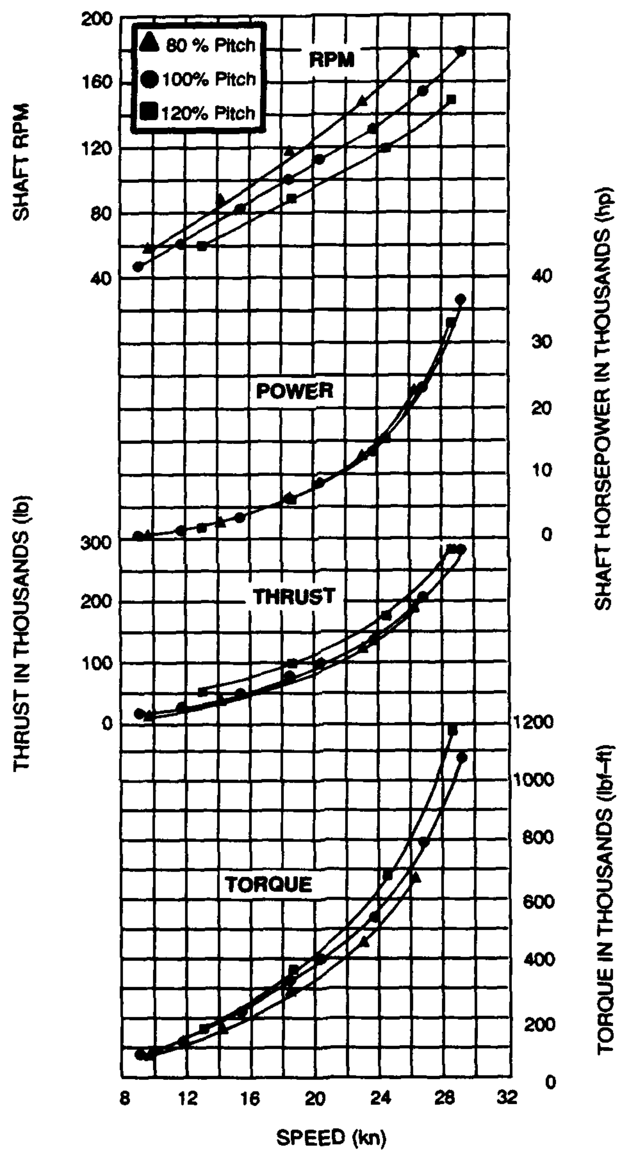


Figure 1. USS OLIVER HAZARD PERRY (FFG 7) Standardization Trials, May 1978.

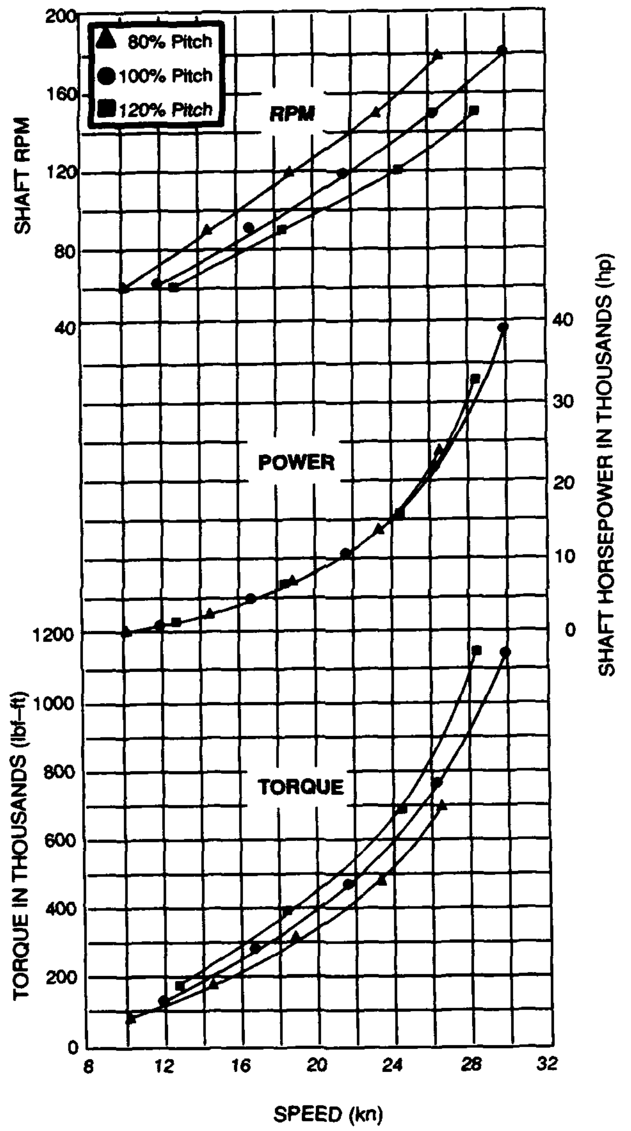


Figure 2. USS UNDERWOOD (FFG 36) Standardization Trials, May 1984.

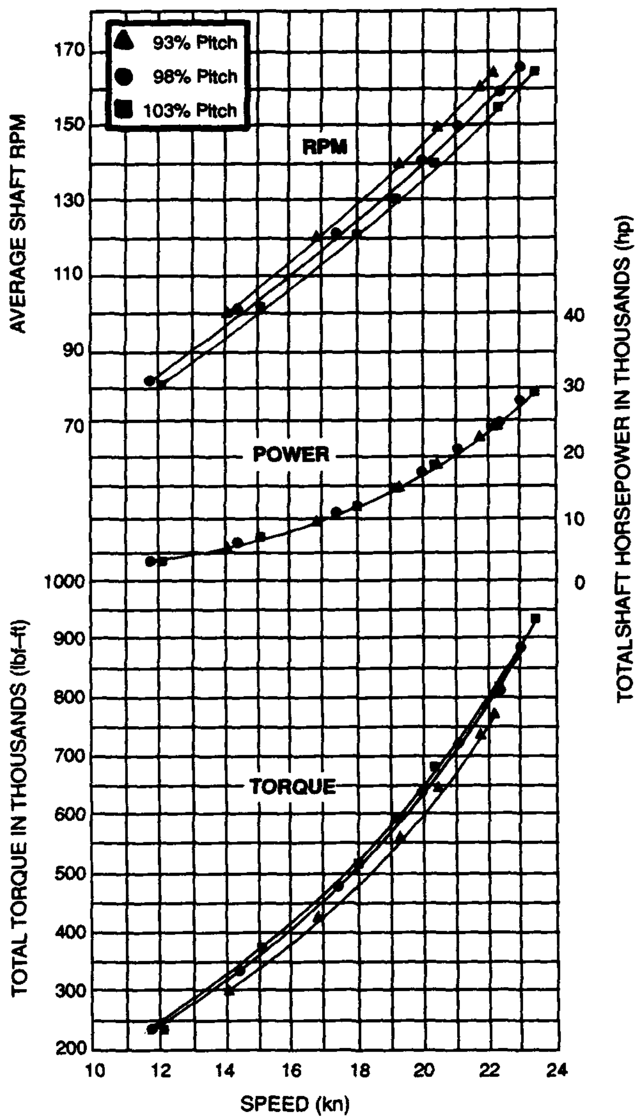


Figure 3. USS WHIDBEY ISLAND (LSD 41) Standardization Trials, March 1985.

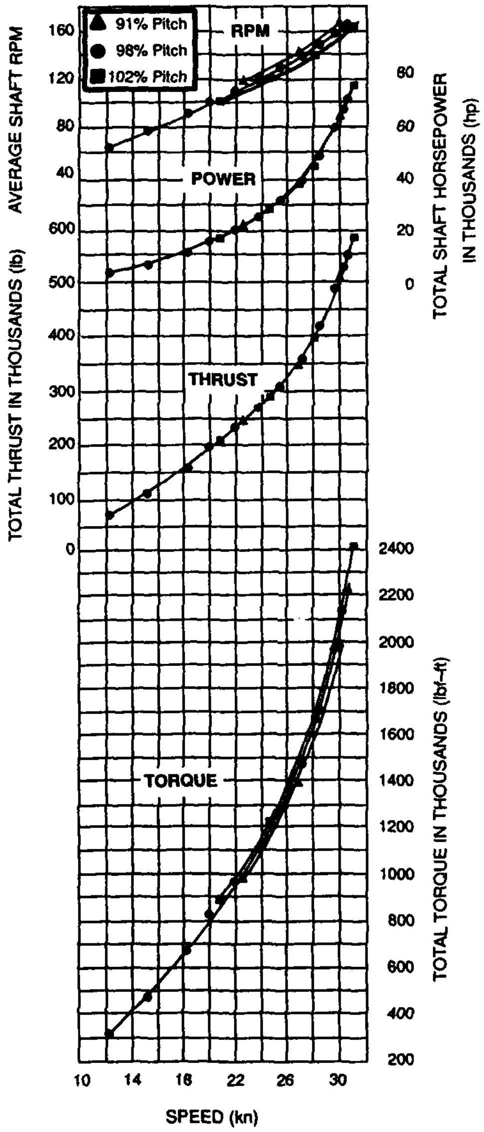


Figure 4. USS VINCENNES (CG 49) Standardization Trials, August 1985.

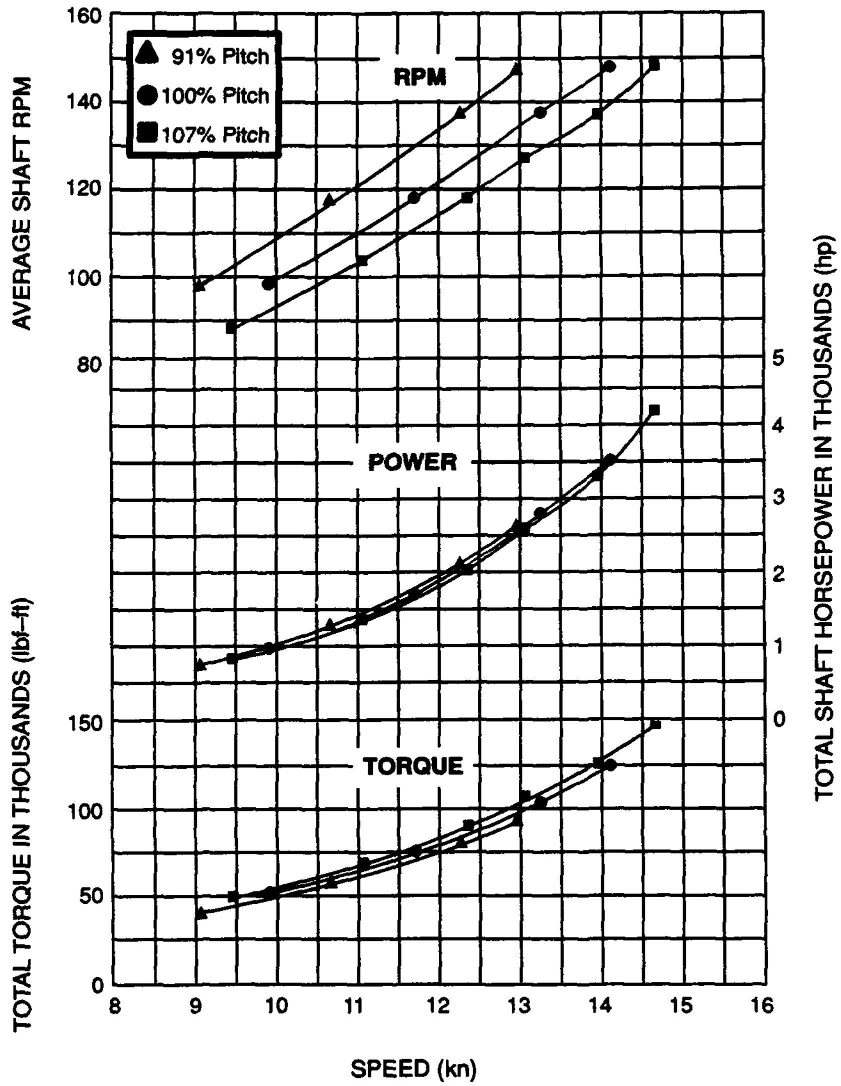


Figure 5. USS SAFEGUARD (ARS 50) Standardization Trials, December 1985.

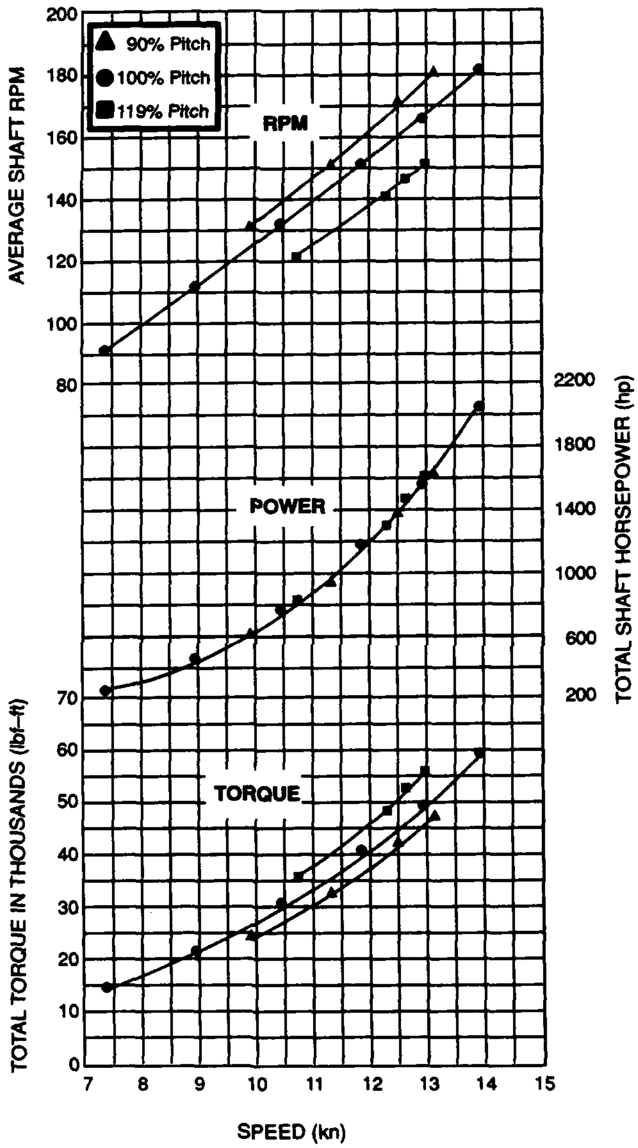


Figure 6. USS AVENGER (MCM 1) Standardization Trials, June 1989.

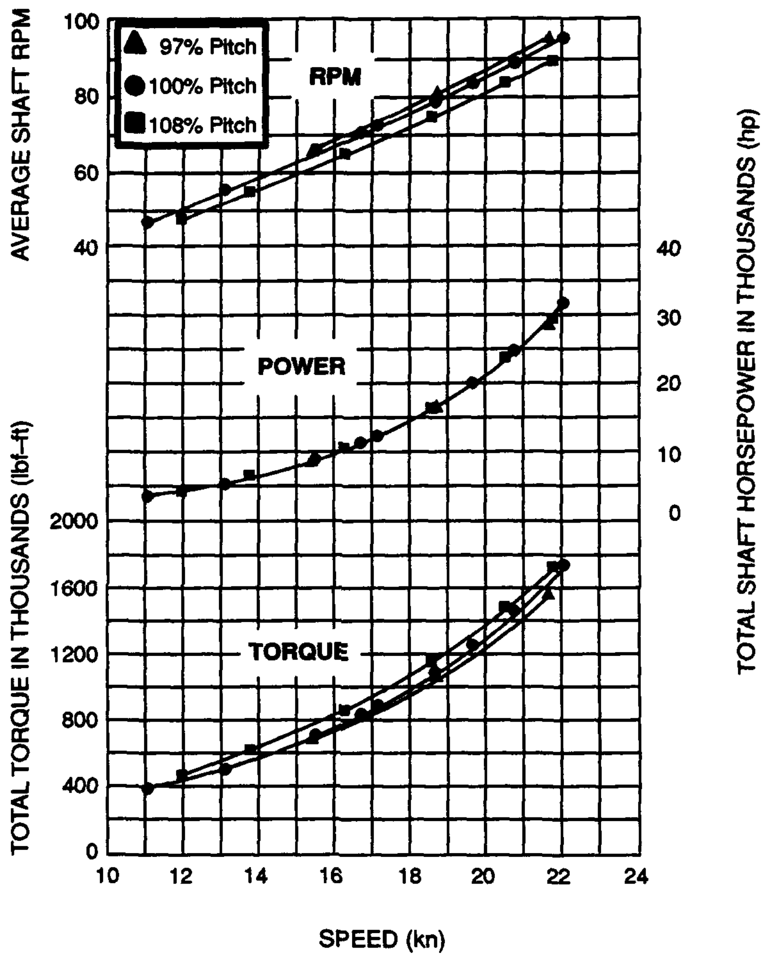


Figure 7. USNS WALTER S. DIEHL (T-AO 193) Standardization Trials, June 1989.

shaft torque, an increase in propeller shaft rpm, a slight decrease in propeller thrust (on one trial) and, in most trials, no change in shaft horsepower. However, on two of the trials, the FFG 7 and FFG 36, where there was a large change in propeller pitch (20%), an increase in shaft horsepower for a constant speed was observed. Table 1 presents representative data taken from these figures at the top speed which was common to all three pitch conditions. Table 2 lists ship trial measurement accuracies for trial data included in the paper.

Table 1. Effect of propeller pitch on ship powering characteristics.

Ship	Ship Speed (kn)	Pitch (%)	rpm	Torque (lbf-ft)	SHP (hp)	Thrust (lbf)
FFG 7	26.25	80	178.2	674,900	22,900	190,500
		100	150.0	726,000	20,730	193,000
		120	131.0	843,000	21,030	212,000
FFG 36	26.50	80	179.8	701,200	24,010	-
		100	152.5	785,000	22,790	-
		120	135.5	870,000	22,450	-
LSD 41	22.15	93	164.8	773,900	24,280	-
		98	158.5	802,000	24,200	-
		103	154.5	821,000	24,150	-
CG 49	29.95	91	169.7	1,997,600	64,550	509,700
		98	162.0	2,040,000	62,920	509,700
		102	156.0	2,110,000	62,670	509,700
ARS 50	12.95	91	147.8	94,000	2,650	-
		100	134.1	97,000	2,480	-
		107	126.0	103,000	2,480	-
MCM 1	12.95	90	178.0	46,000	1,560	-
		100	167.5	49,900	1,590	-
		119	151.2	55,900	1,610	-
T-A0 193	21.65	97	96.0	1,568,600	28,670	-
		100	93.1	1,640,000	29,070	-
		108	89.0	1,715,000	29,060	-

Table 2. Measurement accuracies.

Measurement	Source	Accuracy
Steady Ship Speed	Pulse-Radar System	±0.05 kn
Shaft Torque	Deflection Sensor	±1.5% Full Scale
Shaft Torque*	Deflection Sensor	±2.2% Full Scale
Shaft Speed	Infrared Light Sensor	±0.5 rpm
Shaft Thrust	Load Cells	±3% Full Scale
Propeller Pitch	Ship's Indicator	±2% of Design

* MCM 1 data only.

It should be emphasized that the percent pitch values presented in the figures are nominal values, since they do not reflect adjustments due to thrust compression and temperature changes. The importance of these factors will be explored later in this paper. However, since all of the measurements on a particular trial are made with the same equipment, and under the same trial conditions, the relative change in pitch and its effect on the powering parameters is considered meaningful. Table 3 summarizes the percent change in rpm, torque, shaft horsepower, and thrust for a change in propeller pitch at a constant ship speed. Typically a 1% change in propeller pitch can result in approximately a 1% change in rpm and a 0.5% change in torque. On one trial, a 0.5% change in thrust for a 1% pitch change was observed. On two of the trials, a *0.5% change in shaft horsepower for a decrease of 1%* in pitch was observed. Very small changes in propeller pitch have a direct impact on powering characteristics as shown. Therefore, the importance of being able to accurately determine propeller pitch cannot be overemphasized. This is especially true when taking into consideration model predictions of full scale performance, and full scale/model correlations, where small percentage differences in torque, rpm, and power are very important.

Table 3. Percentage change in ship powering characteristics at a constant ship speed.

Ship	Percent Change in Pitch from Design	Percent Change in rpm	Percent Change in Torque	Percent Change in Power	Percent Change in Thrust
FFG 7	-20	+19	-7	+10	-1
	+20	-13	+16	+1	+10
FFG 36	-20	+18	-11	+5	-
	+20	-11	+11	-1	-

Table 3. (Continued)

Ship	Percent Change in Pitch from Design	Percent Change in rpm	Percent Change in Torque	Percent Change in Power	Percent Change in Thrust
LSD 41	-5	+4	-4	0	-
	+5	-2	+2	0	-
CG 49	–7	+5	–2	+2	0
	+4	–4	+3	0	0
ARS 50	–9	+10	–3	+7	-
	+7	–6	+6	0	-
MCM 1	–10	+6	–8	–2	-
	+19	–10	+12	+1	-
T-A0 193	–3	+3	–4	–1	-
	+8	–4	+4	0	-

3.2. Design Full Power and Design Propeller Pitch

A recurring problem observed on ships with controllable pitch propellers is the inability to develop design full power at design (100%) propeller pitch. Table 4 is a summary of the full power data recorded during the Performance and Special Trials on the previously discussed ships. The table includes the design rpm, torque, and shaft horsepower for each ship. The shaft horsepower developed at the nominal design pitch (100%) is presented as a percentage of design shaft horsepower. For comparison purposes, the pitch closest to the design propeller pitch of 100% is designated as the nominal design pitch. The nominal design pitch and the pitch at maximum power observed are corrected for temperature variations in the propeller system and for propeller shaft thrust compression, with the exception of FFG 7 and FFG 36 data. As can be observed in the table, none of the ships tested reached design full power at design pitch, although the FFG 36 was within 1.3%. Also included in the table is the maximum power observed during the trial and the corresponding pitch. The maximum power observed on two of the ships, ARS 50 and CG 49, occurred at off design pitches of 107% and 108%, respectively, with the ARS 50 actually reaching full power at this condition. It should be pointed out, however, that if this had been a fixed pitch propeller, the ARS 50 would only have delivered 84% of design full power.

Table 4. Full power performance comparisons.

Ship	Design rpm	Design Torque (lbf-ft)	Design SHP	Maximum Power at Nominal Design Pitch (hp)	Nominal Design Pitch (%)	Percent of Design Power	Maximum Power Observed (hp)	Pitch at Maximum Power Observed (%)	Percent of Design Power
FFG 7	180.0	1,167,100	40,000	36,480	101	91.2	36,480	101	91.2
FFG 36	180.0	1,167,100	40,000	39,480	100	98.7	39,480	100	98.7
LSD 41	165.0	1,050,400	33,000	29,340	102	88.9	29,340	102	88.9
CG 49	168.0	2,500,000	80,000	75,940	101	94.9	76,880	108*	96.1
ARS 50	150.0	148,000	4,200	3,530	100	84.0	4,190	107	99.8
MCM 1	180.0	70,000	2,400	2,050	100	85.4	2,050	100	85.4
T-A0 193	95.4	1,791,900	32,500	31,690	102	97.5	31,690	102	97.5

* *This point not plotted on Figure 4.*

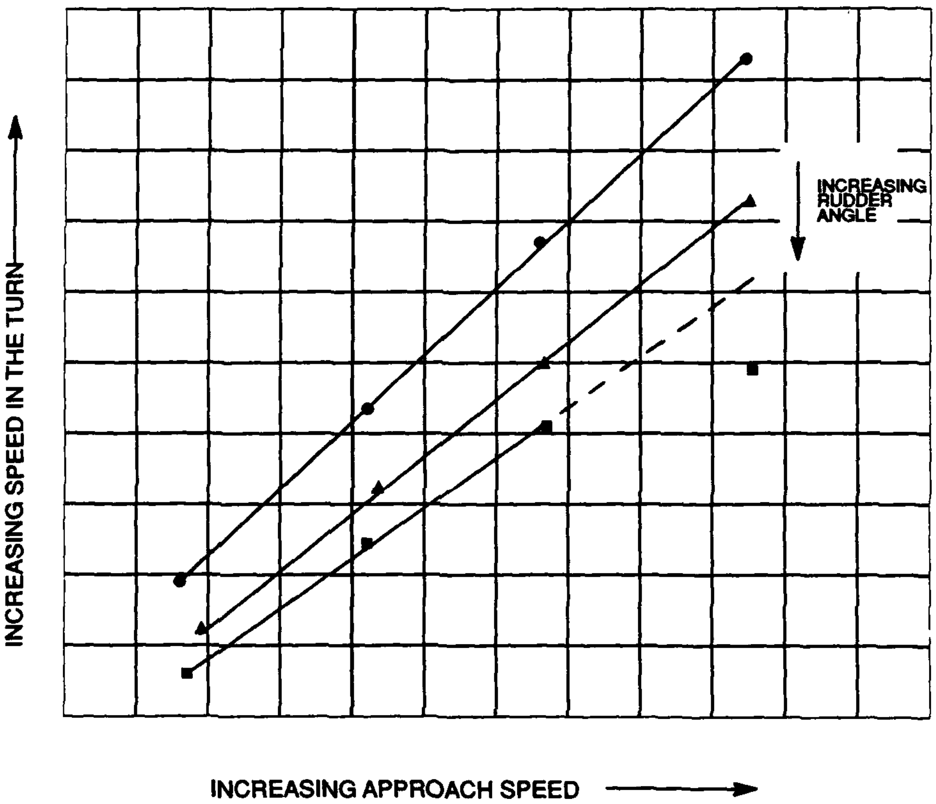


Figure 8. USS VINCENNES (CG 49) speed in the turn.

3.3. Controllable Pitch Propellers and Tactical Circles

A situation which occurred during the trials on the CG 49 is indirectly related to propeller pitch and powering characteristics. The CG 49 automatic control mode is programmed so that propeller rpm is reduced during high speed and high rudder angle turns, to prevent overtorquing of the propeller shaft. This results in a greater reduction of speed in a turn when compared to other slower speed, lesser rudder angle turns. This can be observed in Figure 8 where it can be seen that the flank speed run with full rudder does not fall on the curve. Standard fixed pitch tactical trials are conducted with hands off the engine controls, and typically result in reduced rpm, increased torque, and constant power during the maneuvers. In the case of most controllable pitch systems, when in the automatic mode, the propeller pitch and rpm are held constant in the turn, resulting in an increase of power during the maneuvers as the propeller shafts torque up. This makes the comparison of turning maneuvers from ship to ship and from full scale to model, more difficult.

4. CONTROLLABLE PITCH PROPELLER SYSTEM

The controllable pitch propeller system that is on the ships discussed in this paper is manufactured by the Bird–Johnson Co. The system is made up of three integrated systems: mechanical, hydraulic, and electrical. This section of the paper is broken into a description of each of these systems. The mechanical system of the propeller will be discussed first. This will be followed by a discussion of the hydraulic oil system. Finally, the electrical control system will be discussed.

4.1. Mechanical System

The mechanical system of the propeller can be broken down into three general components: the oil distribution (OD) box, the valve rod, and the hub. A schematic showing the controllable pitch propeller mechanical system with these three components can be seen in Figure 9.

The OD box is forward of the main reduction gear and it houses the auxiliary servomotor. It also has a local pitch indicator where the propeller pitch is mechanically displayed. This mechanical display is directly proportional to the position of the auxiliary servomotor in the OD box. The purpose of the OD box is two-fold: (1) it controls the valve rod by the positioning of the auxiliary servomotor, and (2) it allows the working fluid to pass into the system.

The heart of the system is a long hollow steel tube called the valve rod. This rigid and continuous rod is inside the propeller shaft and it extends from the OD box all the way into the hub. The valve rod has the auxiliary servomotor attached to its forward end and the valve pin attached to its aft end. The valve pin on the aft end of the valve rod rides inside a valve liner which is housed in the hub servomotor. The purpose of the valve rod is to mechanically translate the propeller pitch signal at the OD box into a blade setting at the hub.

The hub consists of the mechanical linkages which cause the propeller blades to rotate to various pitch positions. A more detailed schematic of these components, as well as arrows that designate directions of motion, is shown in Figure 9. These components include the hub servomotor, sliding block, crank ring, and propeller blade. The workhorse in this part of the system is

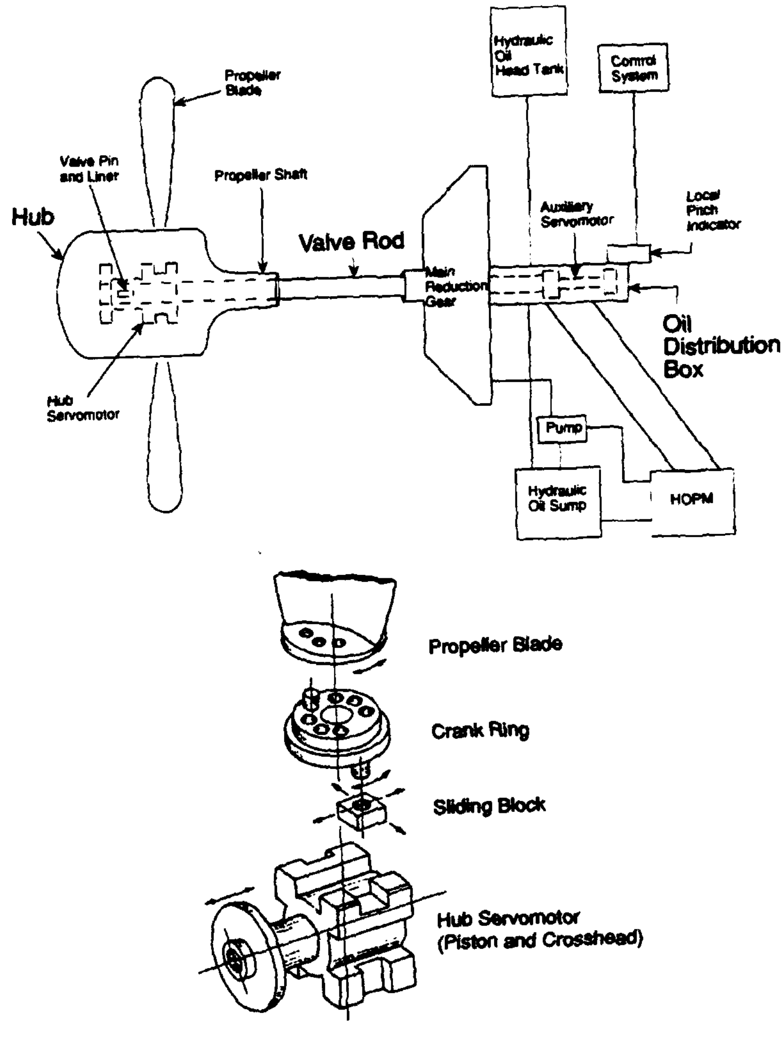


Figure 9. Bird–Johnson controllable pitch propeller system and hub components.

the hub servomotor. The hub servomotor consists of a piston and crosshead which are rigidly attached to each other. The cross head is slotted with each slot corresponding to a particular propeller blade. A sliding block rides in each slot. The sliding blocks have circular holes in their centers where the pin of the crank ring sits. The propeller blades are firmly bolted to their respective crank rings. The purpose of the hub servomotor, along with its associated hub hardware, is to physically move the propeller blades to a particular position.

The mechanical system moves the propeller blades in the following manner. The auxiliary servomotor is moved along the axis of the propeller shaft (axially) to a specific position and is held fast. This axial movement displaces the entire valve rod and valve pin an equal amount of distance. In the hub, when the position of the valve pin changes relative to the valve liner, the working fluid is ported to either side of the piston of the hub servomotor. This causes the piston to move axially. Since the hub servomotor is a rigid body, as the piston moves axially, the crosshead is equally displaced. Displacement of the crosshead causes axial and perpendicular movement of the sliding blocks. Movement of the sliding blocks causes the crank rings to rotate which in turn rotates the propeller blades to various positions. Therefore, axial displacement of the auxiliary servometer is translated into rotational movement of the propeller blade via the valve rod and hub components.

It is important to note that the movement of the auxiliary servomotor will cause the hub servomotor to move. However, the hub servomotor is free to move independently of the auxiliary servomotor. This phenomena will be discussed later in the paper.

4.2. Hydraulic Oil

The working fluid of the controllable pitch propeller system is hydraulic oil. There are four main parts to this hydraulic oil system: the hydraulic oil sump, the Hydraulic Oil Power Module (HOPM), the OD box, and the shaft and hub. The oil in the sump is heated to a nominal operating temperature in the range of 100°F to 130°F. From the sump, the oil is pumped to the HOPM where it is separated into two individual flows: the main servo oil and the auxiliary servo oil. The main servo oil flow is a larger volume and a higher pressure than the auxiliary servo oil flow.

The main servo oil flows constantly through the system. It flows from the HOPM to the aft end of the OD box and then inside the valve rod and down the shaft. At the hub servomotor, the oil is ported to either side of the piston depending on the position of the valve pin and liner. This oil controls the position of the hub servomotor piston which subsequently controls the position of the crosshead. The return oil flows back to the sump between the propeller shaft and the valve rod. However, the main servo oil does not affect the position of the auxiliary servomotor in the OD box.

The auxiliary servo oil flows from the HOPM to the forward end of the OD box via an electro-hydraulic servo valve and then back to the sump. This oil controls the position of the auxiliary servomotor in the OD box.

4.3. Electrical System

The electrical system of the controllable pitch propeller is a feedback control system. The command signal is initiated in the control system and is the signal which represents the ship's ordered pitch. The feedback signal comes from the linear potentiometer which is connected to the auxiliary servomotor at the local pitch indicator. The feedback signal represents the position of the auxiliary servomotor in the OD box. The command signal is continuously compared to the feedback signal. When the signals are different, oil is ported to the necessary side of the auxiliary servomotor to move it to the position where the feedback signal and the command signal are equal. When the two signals are equal, the auxiliary servomotor is hydraulically locked into place.

5. DETERMINING PROPELLER PITCH

This section of the paper will discuss the methods and science involved in determining the propeller pitch. It is divided into two sections. The first section, Physical Pitch Measurements, describes the measurements of the blades relative to the propeller shaft. The Calibration section describes the various pitch settings and temperatures which are necessary to determine a detailed propeller pitch calibration.

5.1. Physical Pitch Measurements

For Performance and Special Trials the pitch/voltage relationship of a controllable pitch propeller is determined prior to the trial. This entails the physical measurement of the position of the propeller blades with respect to a plane normal to the axis of the propeller shaft and the recording of the voltage indicated by the control system. The axial distances from the plane to the leading and trailing edges of each blade at 70% of the radius are measured. The difference between the two measurements is called the axial distance (d) between the leading and trailing edges. The ratio of the axial distance (d) to the blade chord length at the 70% radius (c) is the sine of the pitch angle as shown in Equation 1.

$$\phi = \sin^{-1}\frac{d}{c} \tag{1}$$

Where ϕ = Pitch angle in degrees
d = Axial distance between the leading edge and the trailing edge of the propeller blade at the 70% radius in inches
c = Blade chord length at the 70% radius in inches.

The pitch angle (ϕ) calculated in Equation 1 is entered into Equation 2 to calculate the propeller pitch at the 70% radius.

$$P = 2\pi\ (0.70R)\tan\phi \tag{2}$$

Where P = Propeller pitch at the 70% radius in feet
R = Propeller radius in feet
ϕ = Pitch angle in degrees.

The ratio of this propeller pitch to the design pitch yields the percent propeller pitch.

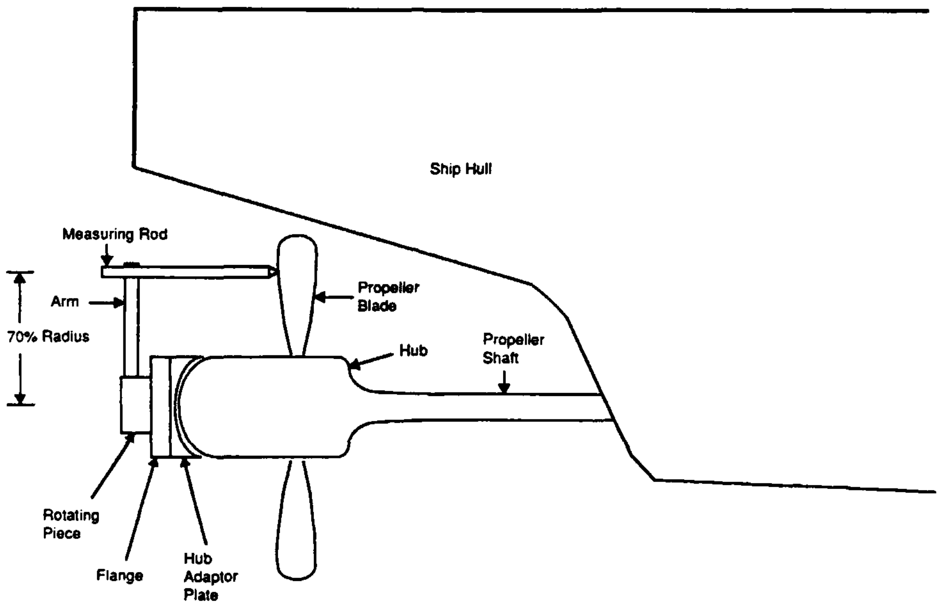


Figure 10. DTRC propeller pitch calibration device in place on a hub.

The device used to make the axial distance measurements was designed and fabricated by DTRC. It is shown attached to the hub in the schematic depicted in Figure 10. It can be fastened to the propeller hub in drydock, or by divers when the ship is waterborne. The device consists of a hub adaptor plate which is firmly fastened to the aft end of the hub. A flange is bolted to the hub adaptor plate. The rotating piece is bolted to the flange. From this rotating piece, an arm extends perpendicular to the propeller shaft. The long rigid measuring rod slides in a small bracket which is attached to the arm at the 70% radius of the propeller. The measuring rod is parallel to the shaft.

5.2. Calibration

The propeller is calibrated at several different pitch settings. At each pitch setting, measurements are taken on all the blades. These measurements are then averaged to yield an average axial distance for that particular pitch setting. This axial distance is used in the above equations to calculate propeller pitch and the percent propeller pitch at each setting. Several marks are normally scribed on the propeller hub and blades. These scribe marks correspond to pitch settings of Full Ahead, Design (100%), Centerline, and Full Astern. Pitch measurements are taken at these pitch settings and any others which might enhance the calibration. Only the pitch/voltage relationship for the ahead pitch settings are used in the calibrations. Figure 11 shows a pitch/voltage curve developed during the USS WHIDBEY ISLAND (LSD 41) starboard pitch calibration. It should be noted that this calibration is conducted in a zero thrust condition. The consequences of this condition will be discussed in a later section of this paper.

The hydraulic oil in the system operates at a nominal design temperature which is subject to variations. These variations will be discussed later in the paper. It is desirable to calibrate for at least three different hydraulic oil temperatures so that pitch readings taken during sea trials can be corrected for oil temperature variations encountered during the trial. The primary or baseline calibration temperature used is one which is representative of the design operational temperature. The other two calibration temperatures are preferably at least 10°F greater and 10°F less than the design operational temperature.

An example of the USS VINCENNES (CG 49) propeller pitch calibration with temperature variations is shown in Figure 12. The plot shows that for a constant signal in the control system, a colder temperature shows an increase in propeller pitch and a warmer temperature shows a decrease in propeller pitch.

6. PHENOMENA WHICH AFFECT PROPELLER PITCH

As implied earlier, propeller pitch should be directly related to the voltage as seen in the control system by the pitch feedback signal. This is not true. When the command signal and feedback signal are equal, the auxiliary servo is hydraulically locked in place. Thus, the pitch is electrically fixed. However, the actual pitch of a controllable pitch propeller is affected by two factors: the temperatures of the individual sections composing the propeller system and the compression of the shaft due to propeller thrust. Each factor may change the position of the valve liner with respect to the valve pin in the hub and hence the propeller pitch.

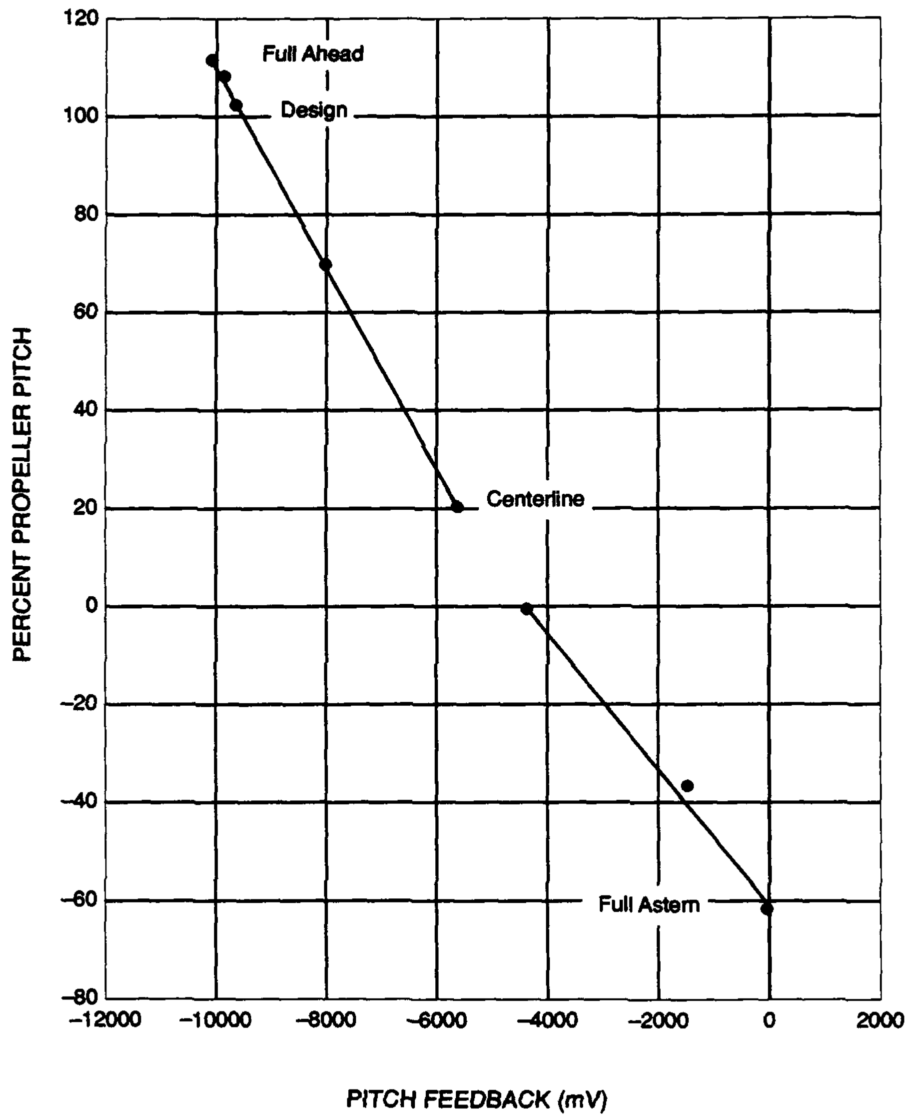


Figure 11. USS WHIDBEY ISLAND (LSD 41) propeller pitch calibration.

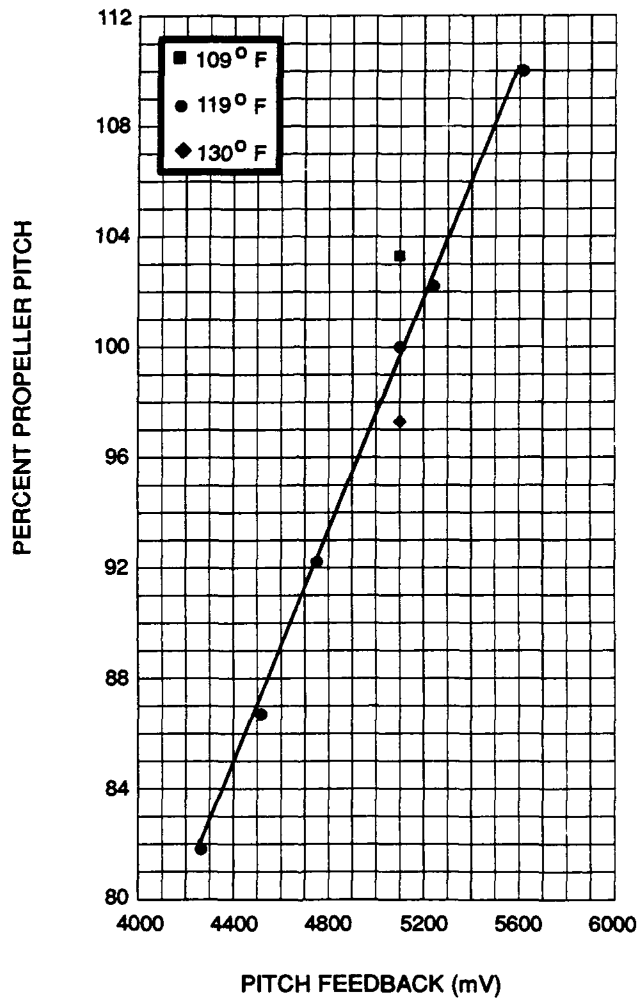


Figure 12. USS VINCENNES (CG 49) pitch calibration with temperature variations.

6.1. Temperature Considerations

The propeller pitch system operates with the hydraulic oil at a nominal operating temperature. The system is also calibrated at this temperature. However, the system temperature can change over time depending on such variables as the hydraulic oil sump heater location, seawater flowing around the shaft and hub outside of the ship hull, ambient temperatures in the compartments where the shafting is located, different cross-sectional areas of shafting, heat from line shaft bearings, and other phenomena. If the system temperature is significantly different from the nominal design operating temperature, the pitch indicated by the feedback signal at the OD box is not the true pitch of the propeller. The actual pitch is affected by the sum of the thermal expansions or contractions seen by the propeller shaft and valve rod.

Thermal expansions or contractions of the valve rod occur in the hub because the auxiliary servomotor on its forward end is hydraulically locked into place in the OD box. Therefore, linear displacements of the valve rod due to thermal expansion or contraction will cause the hub servomotor to move which will change the propeller pitch. Since the ship's ordered pitch has not changed and the auxiliary servomotor is fixed in place, the command signal and the feedback signal will be constant and equivalent. Hence, the pitch indicator will read the ship's ordered pitch which will be a constant value, when in actuality the pitch of the propeller has physically changed at the hub. An increase in system temperature relative to the calibration temperature, will cause a decrease in propeller pitch. Similarly, a decrease in system temperature relative to the calibration temperature will cause an increase in propeller pitch.

It is rather improbable that temperature is or will be monitored in all sections of the shafting (inside or outside of the ship's hull) to develop a picture of how oil temperature affects the relationship between propeller shaft and valve rod in individual locations. In lieu of individual section analysis of this relationship, an overall nominal propeller pitch system temperature is developed based on the average of the hub servo oil temperatures. The temperatures of the oil before it enters the OD box and before it enters the sump are used.

From Figure 12, it can be seen that a one degree change in the overall average temperature of the hydraulic oil system will cause a 0.3% change in propeller pitch on the CG 49. This figure shows that, when the command signal and feedback signal are equal, a hydraulic oil temperature cooler than the nominal operating temperature will yield a greater ahead pitch on the propeller. Therefore, propeller pitch data often have to be adjusted for these temperature variations.

6.2. Propeller Thrust/Shaft Compression Considerations

A direct consequence of the propeller developing thrust is a decrease in propeller pitch. The thrust force is not transmitted to the valve rod housed within the propeller shaft and hub. The thrust generated by the propeller will cause the propeller shaft and hub to compress while the valve rod remains fixed in place. This will cause propeller pitch changes. Since the valve rod is fixed, the electrical system will be constant. However, the mechanical system at the hub will move. This results in the pitch being decreased. The following discussion gives the equations and an example of how thrust changes propeller pitch.

The thrust developed by the propeller of a ship underway results in a compression force on the shaft. This force causes the shaft to compress an amount that can be calculated by using Equation 3.

$$\delta = T/E \times \sum_{i=1}^{N} L_i/A_i \qquad (3)$$

where δ = Propeller shaft compression in inches
T = Propeller shaft thrust in pounds
L = Propeller shaft length in inches
A = Propeller shaft cross-sectional area in square inches
E = Modulus of elasticity in pounds per square inch

This equation shows that the compression is directly proportional to the thrust. It is also dependent on shaft length, cross-sectional area, and shaft material.

WHIDBEY ISLAND (LSD 41) is an example of a multiple shaft ship having varying shaft lengths. Table 5 lists the various shaft sections, their lengths, outside diameters, inside diameters, cross-sectional areas, modulus of elasticity, and length to area ratios. The individual length to area ratios were summed to yield a total length to area ratio of 21.8 in^{-1} and 34.8 in^{-1} for the port and starboard shaft, respectively. The modulus of elasticity of the shaft material is 30,000,000 lb/in^2.

Table 5. USS WHIDBEY ISLAND (LSD 41) propeller shaft characteristics.

	Port	*Starboard*
Shaft Sections, Type I		
Length, in.	1,006	2,061
Outer Diameter, in.	13.75	13.75
Inner Diameter, in.	9.25	9.25
Cross Sectional Area, in^2	81.3	81.3
Summation of length/area, in^{-1}	12.4	25.4
Shaft Sections, Type II		
Length, in.	1,444	1,444
Outer Diameter, in.	18.75	18.75
Inner Diameter, in.	12.50	12.50
Cross Sectional Area, in^2	153.4	153.4
Summation of length/area, in^{-1}	9.4	9.4
Total of length/area, in^{-1}	21.8	34.8
E, modulus of elasticity, lbf/in^2	30.0×10^6	30.0×10^6

No thrustmeter was installed for the WHIDBEY ISLAND (LSD 41) Performance and Special Trials. Based on the full scale shaft horsepower, model powering test data show that for a total shp of 28,300, a thrust of 138,700 lb is generated per shaft.

The maximum shaft thrust will cause the largest compression. The maximum shaft thrust (T_{max}) for WHIDBEY ISLAND is 138,700 lb. This thrust value was used to determine the maximum shaft compressions of 0.101 in. and 0.159 in. for the port and starboard shafts, respectively.

The shaft compression in inches can be related to a change in percent pitch. This is accomplished by taking measurements on the local pitch indicator on the OD box. These measurements show that the control rod moves 0.7188 in. for pitch changes between 90% and 110%. This information is used in Equation 4 to determine the maximum amount of pitch change due to the shaft compression.

$$PC_{max} = \delta_{max} \times \frac{PCR}{m} \tag{4}$$

where PC_{max} = Maximum propeller pitch change in percent
δ_{max} = Maximum propeller shaft compression in inches (from Equation 3)
PCR = Propeller pitch range in percent
m = Movement of valve rod in inches.

The maximum amount of pitch change was determined using Equation 4. The pitch realized was a 4.4% decrease for the starboard propeller and a 2.8% decrease for the port propeller. The preceding examination of the change in propeller pitch due to thrust is shown mathematically in Table 6.

Table 6. USS WHIDBEY ISLAND (LSD 41) pitch change due to shaft compression.

Maximum Shaft Compression

Equation 3. $\delta_{max} = T_{max}/E \times \sum_{i=1}^{N} L_i/A_i$

	Port	Starboard
$\sum_{i=1}^{N} L_i/A_i =$	21.8 in^{-1}	34.8 in^{-1}
$E =$	30,000,000 lb/in^2	30,000,000 lb/in^2
$1/E \times \sum_{i=1}^{N} L_i/A_i =$	7.272×10^{-7} in/lb	1.146×10^{-6} in/lb
$T_{max} =$	138,700 lb	138,700 lb
$\delta_{max} =$	0.101 in.	0.159 in.

Table 6. (Continued)

Maximum Pitch Change Due to Maximum Thrust

Equation 4. $PC_{max} = \delta_{max} \times \frac{PCR}{m}$

	Port	Starboard
$\delta_{max} =$	0.101 in.	0.159 in.
$PCR =$	110% - 90% = 20%	110% - 90% = 20%
m =	0.7188 in.	0.7188 in
$PC_{max} =$	2.8%	4.4%

The determination of pitch change due to thrust is a very dynamic process. The amount of pitch reduction changes as the ship speed increases or decreases. As mentioned earlier in the paper, the initial pitch calibration is done pierside and does not reflect the contribution thrust has on propeller pitch. Without thrustmeter measurements, it is almost impossible to provide a reliable indication of the actual pitch seen at the propeller. Most ships are not instrumented for thrust and as in the case of the WHIDBEY ISLAND, model powering data had to be utilized to characterize the thrust. Full scale trial results are used to refine the model prediction process and consequently, full scale and model data predictions/results are not necessarily always in agreement. Propeller pitch data should reflect the changes due to thrust.

7. CONCLUDING REMARKS

This paper addresses the effects of propeller pitch on a ship's powering characteristics, the inability to attain design full power at design pitch, and the effect of propeller pitch on tactical circle trials. It explains how the controllable pitch propeller system operates and includes a description of the method DTRC currently uses in determining propeller pitch. A discussion of how propeller pitch changes due to temperature variations in the propeller pitch system and shaft compression due to thrust is presented. Coupled with the insensitivity of the feedback control system to these changes in pitch, it is evident that a reliable, accurate in-hub pitch sensing device should be required for all ships equipped with controllable pitch propellers built for the U.S. Navy.

The Center recommended the incporation of such a device in the early stages of the design process on the ARLEIGH BURKE (DDG 51) Class destoryers, and this ship will deliver shortly with an in-hub pitch sensor. The authors look forward with anticipation to the Performance and Special Trials of the ARLEIGH BURKE. Hopefully, we will be able to more accurately determine the actual pitch of the propeller, and its resulting effect on the propulsion characteristics of this new class of destroyer.

REFERENCES

(1) Woo, Everett L., "USS OLIVER HAZARD PERRY (FFG-7) STANDARDIZATION TRIALS", David Taylor Naval Ship Research and Development Center Report, DTNSRDC-79/081, Sep 1979.

(2) Hendrican, Aditha L., and John S. Webb, "USS VINCENNES (CG 49) STANDARDIZATION, TRAILED, AND LOCKED SHAFT TRIALS", David Taylor Naval Ship Research and Development Center Report, DTNSRDC-86/041, Aug 1986.

(3) Sparks, George T., "USS SAFEGUARD (ARS 50) STANDARDIZATION, TRAILED AND LOCKED SHAFT TRIALS", David Taylor Naval Ship Research and Development Center Report, DTNSRDC-86/055, Aug 1986.

(4) Woo, Everett L., and Michael L. Klitsch, "USS WHIDBEY ISLAND (LSD 41) STANDARDIZATION, TRAILED AND LOCKED SHAFT TRIALS", David Taylor Research Center Report DTRC-89/031, Dec 1989.

(5) Klitsch, Michael L., and Lawrence A. Rogers, "USNS WALTER S. DIEHL (T-AO 193) STANDARDIZATION TRIAL RESULTS", David Taylor Research Center Report, DTRC-90/001, Jan 1990.

(6) Boboltz, David A. Jr., "STANDARDIZATION, LOCKED SHAFT, AND TRAILED SHAFT TRIALS ON USS AVENGER (MCM 1)", David Taylor Research Center Report DTRC-90/002, Mar 1990.

A MAN-MACHINE SYSTEM APPROACH TO MODEL VESSEL TRAFFIC

by D. ten Hove* and P.H. Wewerinke*,**

* Maritime Research Institute Netherlands (MARIN)
** University of Twente

1. ABSTRACT

This paper deals with a model of a large scale man-machine system. In general this implies a multifunction, multicrew process, implying interrelated subsystems. In the paper it is assumed that only the subsystems are interrelated. For this reason human operators have to estimate the state of their own subsystem and of all pertinent other subsystems, and the relationships between them. This nonlinear filter problem is solved by means of linearized and extended Kalman filters. Based on these estimates, human operators control their own subsystem and decide and react to avoid unacceptable subsystem interference.

The model is applied to the concrete problem of vessel traffic control. This implies a number of ships in a confined area. The navigation of each ship is based on a planned route. In addition, collision avoidance is modelled. This involves perception, (nonlinear) estimation, decision making and standardized control. Also the supervising role of a vessel traffic service is considered.

2. INTRODUCTION

Many maritime design and operational problems involve the human navigator and helmsman. Therefore, it is necessary to include the relevant human factors in the study of these problems. Typically, it concerns a complex large scale process consisting of many variables such as ship dynamics, environmental variables (visibility, disturbances), navigational aids, task definitions, human operator functioning, etc.

In order to obtain meaningful solutions to these problems, it is necessary to investigate these problems systematically including all relevant factors and their mutual relationships. Also the complexity of manned large scale systems requires a systematic approach to describe the components of the total system and their mutual interaction.

An approach to study these problems is the use of fast-time simulation models in which all relevant components are modelled together with their interactions. Especially early in the design stage, such models can be used to analyze systematically all relevant factors of the complex problem and to select design alternatives.

In this paper a mathematical model is discussed describing the complex large scale dynamic man-machine system applied to the vessel traffic. The model describes the total vessel traffic control process including the role of the human operator, (HO), as the human factor plays a dominant role in the complex vessel traffic process.

The model contains a number of ships, navigating in a given confined area, with a given planned route. Apart form this normal operation, collision avoidance is modelled, i.e. the detection of a possible conflict and the subsequent actions taken by the ship(s) involved. Also the vessel traffic services (VTS) is modelled in it's possible role of monitor, conflict detector and advisor of the total vessel traffic system.

The model of the vessel traffic process must contribute to answering questions related to: safety, in terms of statistical measures of relative ship positions, the effect of HO functioning on safety, necessary information to perform the tasks (by the crews of the ships and the VTS), communication between ships and VTS, optimization of procedures, automation issues of the vessel traffic process etc. A detailed description of the model is given in Chapter 3.

3. MODEL STRUCTURE

3.1 General

In this chapter the model structure of the vessel traffic control process is presented.

Generally, manned large scale systems typically imply a multifunction, multicrew process, consisting of interrelated subsystems. A block diagram of the components and their interrelationships is presented in Fig. 1. As can be seen different tasks can be performed by different HO, being related to different subsystems. Furthermore the subsystems can be more or less coupled, as described by the interrelation function. The HO again are controlled by a supervisor. Both HO and supervisor derive information from the subsystems by using instruments, outside view and personal communication. Besides the information derived from the system, HO and supervisor can use information from the rule base, consisting of procedures, rules and regulations.

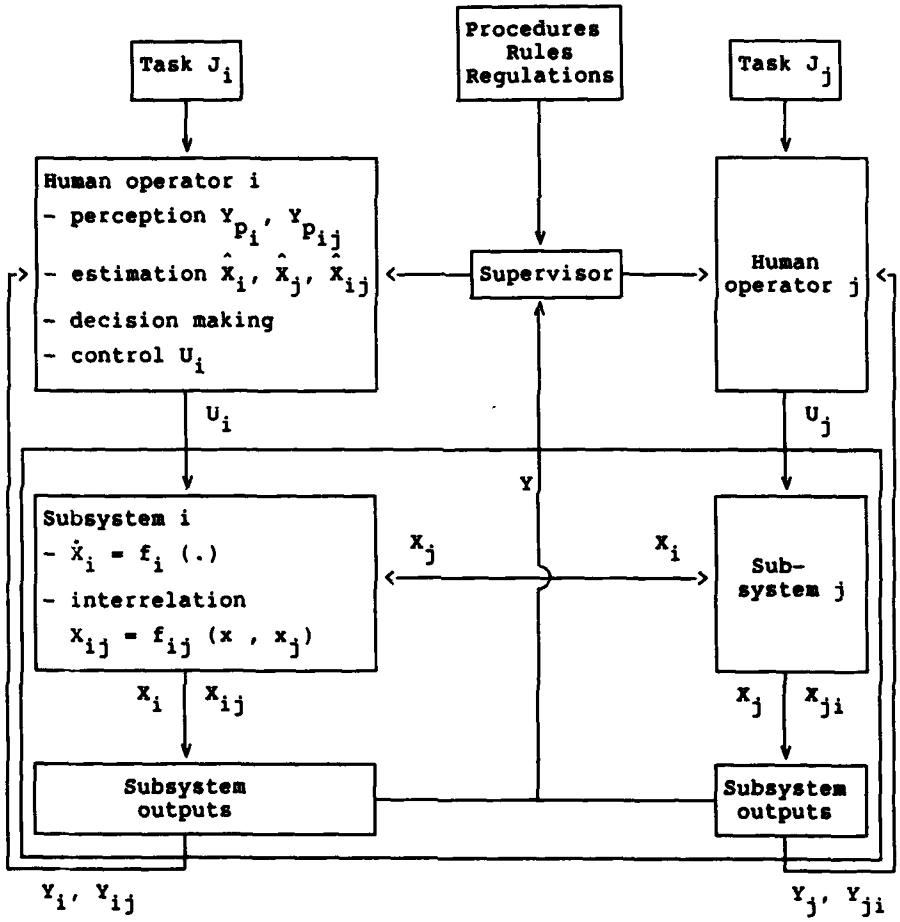


Figure 1. Block diagram large scale man-machine system

The foregoing general model structure is applied to the concrete problem of vessel traffic control. The ships in the traffic area are identified with the subsystems, the navigators with the human operators and the VTS with the supervisor. A block diagram of this situation is presented in Fig. 2.

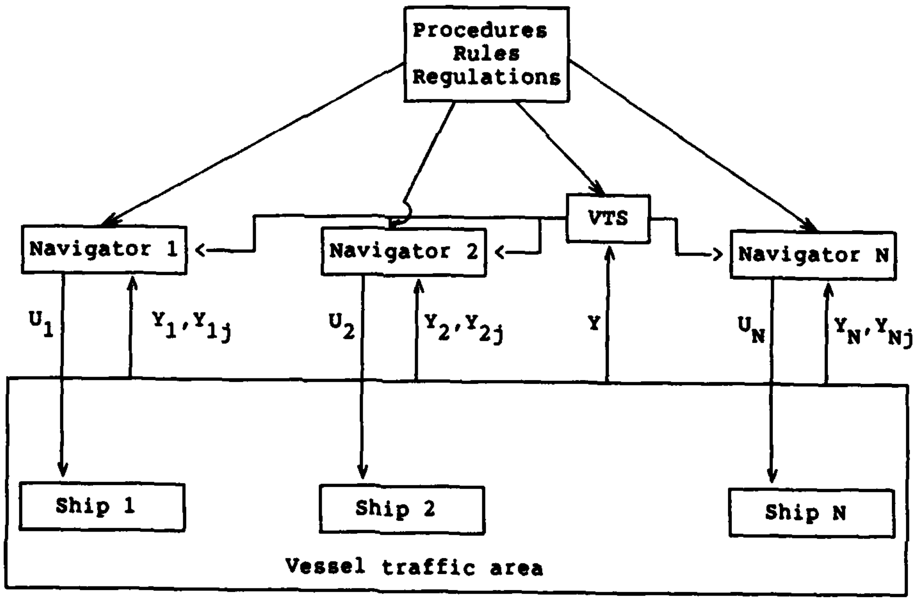


Figure 2. Block diagram of the vessel traffic model

This implies a number of ships, with a given destination in a confined area. The navigation of each ship is based on a planned route, consisting of straight tracks, which is updated via information of the visual scene, containing aids to navigation, instruments and the vessel traffic services (VTS), important disturbances are current and wind.

Normal operation amounts to tracking the planned route. Abnormal operations involves the detection of a possible conflict, i.e. a collision or grounding, and the subsequent actions taken by the ship(s) involved, i.e. collision avoidance manoeuvres. Both the collision situations and the subsequent avoidance manoeuvres are strongly determined by rules and procedures.

The collision or grounding risk is recognized on board by the HO, but also the VTS is considered in its role of monitor, conflict detector and advisor of the total vessel traffic system.

The ultimate criteria of the traffic process are safety and economy. Derived measures or these are collision and grounding risk (probabilities) and traffic flows (densities and travel times). These are related to the following aspects: ship dynamics, and their planned routes, number of ships, on board navigation instruments, geometry of the traffic area, visibility conditions, navigational aids, environmental conditions, HO functioning, information available to the VTS, the role of the VTS, communication between HO and VTS and rules and regulations. These aspects are included in the model presented.

3.2 Ship dynamics

In general, the N ships are assumed to be nonlinear and can be described by a nonlinear system model. In our case, assuming no hydrodynamic coupling, the ship dynamics are independent, and the ship models are given by:

$$\dot{\underline{X}}_i\ (t) = f_i\ (\underline{X}_i(t),\ \underline{U}_i(t),\ \underline{W}_i\ (t),\ t) \tag{3.1}$$

$$\underline{Y}_i\ (t) = g_i\ (\underline{X}_i(t),\ \underline{U}_i(t),\ t) \tag{3.2}$$

$$\underline{Y}_{ij}(t) = g_{ij}(\underline{X}_i(t),\ \underline{X}_j(t),\ \underline{U}_i(t),\ t) \tag{3.3}$$

$i = 1, \ldots, N$

$j = 1, \ldots, i-1, i+1, \ldots, N$

with state $\underline{X}_i \in \mathbb{R}^{n_i}$, control $\underline{U}_i \in \mathbb{R}^{k_i}$, system outputs $\underline{Y}_i \in \mathbb{R}^{m_i}$, $\underline{Y}_{ij} \in \mathbb{R}^{m_{ij}}$ and $\underline{W}_i \in \mathbb{R}^{l_i}$ represent a Gaussian white noise process with power spectral density matrix $R_i(t)$.

The nonlinear ship dynamics can be represented in a simplified form assuming no drift (lateral ship velocity) yet describing the main response characteristics (Ref. 2). Referring to eq. (3.1), the resulting vectors are (dropping for the moment the subscript i, indicating ship i, and the index t, indicating time dependency)

$$\underline{X} = \mathrm{col}(U, R, \Psi, X, Y)$$

$$\underline{U} = \mathrm{col}(\Delta U_c, \delta)$$

$$\underline{W} = \mathrm{col}(W_1, W_2) \tag{3.4}$$

$$f = \begin{bmatrix} a\,U & + b\,\Delta U_c & \\ -\frac{1}{T} R & + \frac{K}{T} \delta & \\ R & & \\ U\cos\psi & & + W_1 + \dot{X}_s \\ U\sin\psi & & + W_2 + \dot{Y}_s \end{bmatrix}$$

where U is the longitudinal speed relative to the water, R is the rate of turn, ψ is the heading, X and Y are the earth-fixed coordinates, ΔU_c is the commanded speed change, δ is the rudder angle, $W_{1,2}$ represent the random system disturbances with constant components $\dot{X}_s$ and $\dot{Y}_s$.

The standard procedure is followed to describe the nonlinear dynamic system behaviour ($\underline{X}$) in terms of a state reference ($\underline{X}_0$) and deviations $\underline{x}$ form this reference; thus $\underline{X} = \underline{X}_0 + \underline{x}$, etc. This linearization scheme yields a time-varying reference model and a time-varying linear system description.

It is assumed that the navigator may observe variables provided by instruments (radar, compass, log, etc.) and by the visual scene (buoys, leading lights, conspicuous points, distance a_{ij} and angle of view ψ_{ij} to an other ship, etc.).

3.3 The navigator

In the model the navigators are assumed to be involved in perception, attention allocation and information processing, to control the process as specified by his task and determined by the rules and regulations. In this context control has a broad meaning, involving planning, sequential decision making and compensation for unpredictable effects.

a. Task. The task considered for each navigator is to achieve a desired trajectory (tracking task) in some optimal sense, i.e. controlling the state $\underline{X}$ over some fixed interval of time $[0,T]$ by realizing a control history $\{\underline{U}(t),\ t \in [0,T]\}$ which minimizes the cost functional

$$J_i(U) = E\{(\underline{X}(T) - \underline{X}_d(T))^T Q_x(T) (\underline{X}(T) - \underline{X}_d(T)) + \\ + \int_0^T [(\underline{X}(t) - \underline{X}_d(t))^T Q_x(t) (\underline{X}(t) - \underline{X}_d(t)) + \\ + \underline{U}^T(t) Q_u(t) U(t)]dt\} \tag{3.5}$$

where $\underline{X}_d$ indicates the desired trajectory and Q_x and Q_u are weighting matrices. This optimal control problem is solved in Ref. 1.

b. Perception. It is assumed that a navigator derives information about his ship from instruments, the outside world and personal communication. This is described by the vector functions g_i. In addition, the navigator generally does not know the control inputs of the other ships. However, the assumption is that the navigator i can perceive quantities that are related to both his own ship i and to one other ship j. This is described by the vector functions g_{ij}, $i=1, \ldots, N;\ j=1, \ldots, i-1, i+1, \ldots, N$.

The system outputs are perceived with a given inaccuracy. To allow for intermittent observations, perception is described in discrete time (again dropping the index i)

$$\underline{Y}_p(t_k) = \underline{Y}(t_k) + V(t_k) \tag{3.6}$$

with $V(t_k)$ an independent, Gaussian, white noise process with spectral density matrix P that is dependent on the output magnitude, perceptual threshold and navigator attention (Refs. 1 and 3).

In eq. (3.6) it is assumed that the navigators internal time delays associated with perceptual, central processing, neuromotor pathways and communication and transport delays are negligibly small compared with the process time constants. Otherwise, a pure time delay can be assumed in eq. (3.6), for which delay the navigator has to compensate to obtain an estimate of the present state.

c. Estimation. Based on the perceived information the navigator has to make an estimate of (part of) the system state. In general, however, each navigator will have to estimate not only his own ship state $\underline{x}_i$, but also other relevant ships $\underline{x}_j$. In addition, it is assumed that a navigator makes decisions, using decision variables based on given relationships between ships. The relationship between ship i and j will be indicated with $\underline{x}_{ij}$ and will have to be estimated because $\underline{x}_{ij}$ is a stochastic process, related to $\underline{x}_i$ and $\underline{x}_j$.

The reason for distinguishing between the estimation of $\underline{x}_i$, $\underline{x}_j$ and $\underline{x}_{ij}$ is that each category is typified by different conditions, which require different procedures to describe the nonlinear estimation process.

More specifically, it is assumed that the navigator knows the reference state of his own ship. Thus, the estimation of his own ship state $\underline{x}_i$ can be described in terms of a Kalman filter of the linearized system model. The linearization scheme yields a time-varying reference model (assumed to be known by the navigator) and a time-varying linear system model given by:

$$\dot{\underline{x}}_i(t) = A_i \, \underline{x}_i(t) + B_i \, \underline{u}_i(t) + E_i \, \underline{W}_i(t) \tag{3.7}$$

$$\underline{y}_i(t_k) = C_i \, \underline{x}_i(t_k) + D_i \, \underline{u}_i(t_k) \tag{3.8}$$

where A_i is the Jacobian matrix of f_i with respect to $\underline{x}_i$ (the state transition matrix), etc. For the (standard) filter equations the reader is referred to Ref. 4. The result, which is based on the assumption that the navigator knows the ship dynamics, the control U_i and the noise covariances R_i and P_i, is an estimate of $\underline{x}_i$ given by $\underline{x}_i = \underline{x}_{0_i} + \hat{\underline{x}}_i$

The estimation of the other ship states $\underline{x}_j$, by navigator i, can not be treated in a similar way if it is assumed that the navigator of ship i does not know the nominal behaviour of ship j. In other words, it is not possible to specify a reference state and follow the foregoing linearization scheme.

There are many approaches to solve this nonlinear filtering problem, all involving approximations of the optimal nonlinear filter. Moreover, there does not seem to be a straightforward way to make a theoretical comparison of the estimation qualities of the various nonlinear filter techniques. For this reason, in this paper a minimum variance estimation procedure is followed, corresponding with the conditional mean (Ref. 4). A maximum-likelihood estimator

could be considered, because the nonlinear system is generally not Gaussian, but the resulting optimal nonlinear filter will have to be approximated, leading to similar results as obtained with the minimum variance procedure (Ref. 5). The resulting filter equations are known as the Extended Kalman filter.

In filtering the other ship states there are several ways to adapt for the unknown control inputs $\underline{U}_j$. One can increase the system disturbance covariance so as to increase the filter gain and emphasize the observations, one can estimate $\underline{U}_j$, or one can do both (Refs. 6 and 7).

The third category of estimates concerns the decision variables $\underline{X}_{ij}$ that describe the relationships between ships. These variables are given by:

$$\underline{X}_{ij}(t) = f_{ij}\ (\underline{X}_i(t),\ \underline{X}_j(t)) \tag{3.9}$$

These nonlinear relationships imply a non-Gaussian probability distribution of $\underline{X}_{ij}$. Instead of trying to find (approximated) filter equations based on the conditional probability distribution, in this paper the approach is used to derive stochastic differential equations for $\underline{X}_{ij}$ and to obtain a minimum variance estimate of $\underline{X}_{ij}$ in terms of an Extended Kalman filter. This is the same approach as taken for the estimation of the other ship states $\underline{X}_j$.

<u>d. Control and decision making.</u> It is assumed in this paper that, in normal operation, the navigator controls his own ship given by eq. (3.1)-(3.3) and (3.4) by minimizing J, given by eq. (3.5). This represents a standard stochastic LQG-control problem. The solution is contained in many references (e.g. Ref. 8).

In case the track X_d is relatively constant, one can consider to describe the control process on the basis of the steady-state solution. This results in a feedback control given by

$$\underline{u}_i = -F\ \hat{\underline{x}}_i \tag{3.10}$$

with F the feedback gain matrix.

Finally, it is assumed that decisions are made based on the decision variables described by $\underline{X}_{ij}$. Many decisions can be formulated in terms of a (multiple) comparison of $\underline{X}_{ij}$ with a reference ("if $\underline{X}_{ij}$ exceeds a given value, then ...").

As the probability distribution of $\underline{X}_{ij}$ is generally unknown and not Gaussian, it is not possible to construct a likelihood ratio test. However, a reasonable approach to describe navigator behaviour is to test estimated values of $\underline{X}_{ij}$ against corresponding thresholds.

These threshold values are now model parameters, which may be selected based on task considerations or prescribed rules. Both the navigators and the VTS may be involved in this type of decisions. Examples of the decision rules and procedures for the possible subsequent actions are given in the next chapter.

3.4 Collision avoidance

During navigation in congested waters, a principal task of the navigator is to avoid collisions with other ships or fixed objects. For this purpose, the navigator has to observe his environment in order to recognize in time the occurrence of an encounter with (e.g.) an other ship. Based on the "Rules of the Road" of the International Maritime Organization (Ref. 9) and referring to (Ref. 10), the encounter situation and the required collision avoidance can be structured in the following way, as reflected by in the procedures, rules and regulations.

An encounter is defined as the situation in which all following variables are smaller than a *reference value:*

1. The distance a_{ij} between ship i and j.
2. The closest point of approach (CPA) c_{ij}. This is defined as the distance between the vector of the relative velocity (between the ships) and ship i.
3. The time T_{ij} to reach the closest point of approach.

The mathematical relationships between these variables and the state of the ships involved are derived in (Ref. 6) and clarified in Fig. 3.

The result is presented in the form of eq. (3.9)

$$\underline{x}_{ij} = \mathrm{col}(a_{ij}, c_{ij}, T_{ij}) \tag{3.11}$$

$$f_{ij} = \begin{bmatrix} \left((X_j - X_i)^2 + (Y_j - Y_i)^2\right)^{\frac{1}{2}} \\ \dfrac{(U_j\cos\Psi_j - U_i\cos\Psi_i)(Y_j - Y_i) - (U_j\sin\Psi_j - U_i\sin\Psi_i)(X_j - X_i)}{(U_j^2 + U_i^2 - 2U_jU_i\cos(\Psi_j - \Psi_i))^{\frac{1}{2}}} \\ \dfrac{(U_j\cos\Psi_j - U_i\cos\Psi_i)(X_j - X_i) - (U_j\sin\Psi_j - U_i\sin\Psi_i)(Y_j - Y_i)}{U_j^2 + U_i^2 - 2U_jU_i\cos(\Psi_j - \Psi_i)} \end{bmatrix} \tag{3.12}$$

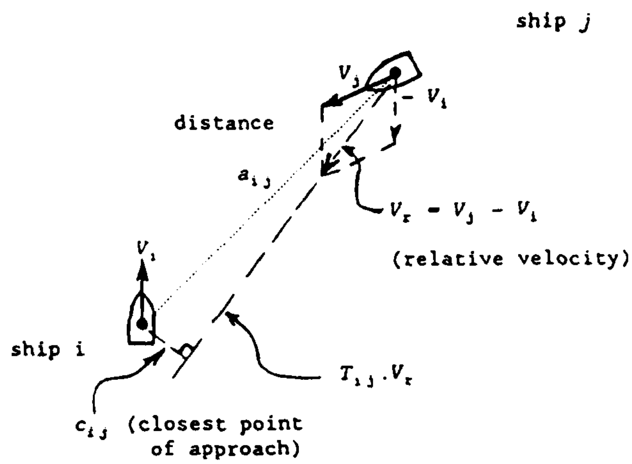


Figure 3. Geometry of an encounter

So we have an encounter, if all elements of $\underline{X}_{ij}$ are smaller than the *corresponding elements of the criterion* $\underline{X}_{c_{ij}}$.

Three types of encounters can be distinguished, each requiring a specific avoidance action:

1. **meeting;** both ships are burdened (to make an evasive manoeuvre to starboard, *corresponding with a given lateral displacement*);

2. **overtaking;** the overtaking ship is burdened (to realize a given lateral displacement), the other ship is privileged (having right of way, maintaining the course and speed);

3. **crossing;** the starboard ship is privileged, the port ship is burdened. If possible, the evasive manoeuvre is towards starboard otherwise a port manoeuvre must be made. The manoeuvre corresponds to a given heading change and a given lateral displacement.

The precise classification is depending on the relative positions and orientations of both ships. For details the reader is referred to Refs. 1 and 10. Although both the encounter situation and the appropriate response may involve more than two ships, it is assumed in this paper that a collision avoidance situation can be described as an encounter of two ships (i and j) at the time. The situation that more than two ships are involved is considered as a sequence of encounters between two ships. Discussions with nautical experts support such an approach.

The navigator is assumed to decide about the occurrence of an encounter by comparing $\underline{X}_{ij}$ with $\underline{X}_{c_{ij}}$.

However, because $\underline{X}_{ij}$ is a stochastic process, he is using an estimate of X_{ij}. This estimate is compared with the criterion value $\underline{X}_{c_{ij}}$. If all elements of $\underline{X}_{ij}$ are smaller than the corresponding criterion value the decision D_1 is made, followed by an action if ship i is burdened. Thus

$$\hat{\underline{X}}_{ij} \underset{D_1}{\overset{D_0}{\gtrless}} \underline{X}_{c_{ij}} \tag{3.13}$$

The evasive manoeuvre is characterized by a given lateral displacement and a given (specified or reasonable) heading change. This standard manoeuvre is uniquely realized by a bang-bang control sequence with a given maximum rudder angle. The switching times are determined by the (linearized) ship dynamics. For details the reader is referred to Ref. 6. It is assumed that the evasive manoeuvre is followed by a symmetric manoeuvre to resume the originally planned route.

3.5 Vessel traffic services

In congested areas (rivers, ports, etc.) a VTS can be helpful to minimize the risk of collisions. Although presently a VTS normally plays only an advisory role (only after the occurrence of an accident a VTS can give commands) its role may change in the future, comparable to the air traffic control development. At any rate, it will be useful to guide and support such a development with a model of vessel traffic control, in which the role of the VTS may include monitoring and conflict detecting to advise or command the total vessel traffic system.

The simplest way to model a VTS is to assume that the navigator receives given (extra) observations (from the VTS). These observations will affect the estimating process and, therefore, the traffic process. Any communication uncertainty can be accounted for in terms of the observation noise level.

A more advanced role of the VTS can be modelled by assuming that the VTS will have the information to make an estimate of the total vessel traffic process and use this to detect any conflict. Based on this the VTS can feedback any advice or command to the navigator(s). It can be assumed that this feedback is taken into account with a given time delay. This approach will increase the model complexity considerably (not conceptually, as the same model elements as before will be involved).

4. MODEL CAPABILITY AND APPLICATIONS

The vessel traffic model is nonlinear because of the nonlinear differential equations and the nonlinear estimation process. Therefore, no closed form expressions can be derived for statistical measures, such as collision probabilities. Thus the model must be used for time (Monte Carlo) simulations. For example, for typical (crucial or interesting) configurations time simulations can be made. The resulting trajectories can be considered and combined to obtain measures for collision probabilities and traffic flows. In addition, measures will be available for the effect of visual informatial variables, or the effect of rules and procedures, on system performance and measures of navigator behaviour related to visual scanning, situation uncertainty and workload.

The model can be applied to a variety of vessel traffic problems. It provides the structure to analyze the effect on safety and traffic handling of (among others) the following variables: ship dynamics, on board navigation instruments, visibility and environmental conditions, aids to navigation, navigator functioning, number of ships and routes in the traffic area, procedures and rules, role of the VTS, etc.

5. CONCLUDING REMARKS

In this paper a mathematical model is discussed that deals with the complex, large scale vessel traffic control system including human operator functions. This implies a number of ships in a given confined area. The navigation of each ships is based on a planned route, which is updated via information of the visual scene, instruments and vessel traffic services.

Both normal operation and conflicts behaviour is modelled. The latter involves the assessment of collision risk and the execution of collision avoidance manoeuvres. The VTS is considered in its role of monitor, conflict detector and advisor of the total vessel traffic system.

REFERENCES

(1) P.H. Wewerinke, "Models of the Human Observer and Controller of a Dynamic System", Ph.D. Thesis, University of Twente, 1989.

(2) J. van Amerongen, et al., "Mathematical Modelling of Ships", Proceedings of the 4th Ship Control Systems Symposium, Vol. 4, pp. 163-173, The Netherlands.

(3) D.L. Kleinman and S. Baron, "Manned Vehicle System Analysis by Means of Modern Control Theory", NASA CR-1753, 1971.

(4) A. Gelb, (ed.), "Applied Optimal Estimation", MIT press, Cambridge, MA., 1974.

(5) R.E. Mortenson, "Maximum-Likelihood Recursive Nonlinear Filtering", Journal of Optimization Theory and Applications, Vol. 2, No. 6, November 1968.

(6) W.I. van der Ent, "Modelling Vessel Traffic", Thesis (in Dutch), University of Twente, The Netherlands, 1989.

(7) A. Moghaddamjoo, "Approaches to Adaptive Kalman Filtering", Journal of Control Theory and Advanced Technology, Vol. 5, No. 1, pp 1-18, Mitra Press, 1989.

(8) J.S. Meditch, "Stochastic Optimal Linear Estimation and Control", Mc Graw-Hill, 1969.

(9) International Regulations for Preventing Collisions at Sea ("Rules of the Road") 1972, I.M.O. 1973, In: Lloyd's Nautical Yearbook, pp. 302-335, 1987.

(10) W.J. van Meurs, et al., "Observation Related Port Approach Traffic Simulation (ORPATS), I: General and Nautical Aspects of Computerized Marine Traffic Simulation", Report KSLA, AMSR. 0048.77, 1978.

(11) P.H. Wewerinke, et al., "Model of Large Scale Man-Machine Systems with an Application to Vessel Traffic Control", Proceedings of the 1989 IEEE International Conference on Systems, Man and Cybernetics, Cambridge, Massachusetts, November 1989.

FUNCTIONAL AND PERFORMANCE ANALYSIS OF GENERIC SCC

by A M Pechey Ph D, C Psychol, AFBPsS
Caversham Consultants, Reading, UK*
and M I Hawken BSc, CEng, MIEE
Sea Systems Controllerate, Ministry of Defence (UK)

1. ABSTRACT

This paper describes one of a series of studies carried out by MOD(PE) Sea Systems Controllerate and Electronic Facilities Design (EFD) Ltd in support of the specification of the next generation of Ship Control Centre (SCC) in the Royal Navy. The major purpose of the study is to investigate if and how improvement can be achieved in operational effectiveness by increased automation of SCC functionality. Data collection involved surveying relevant publications, a structured interview programme and observation of training exercises. Data was collected on the tasks currently carried out to support SCC functions (in a number of different classes of RN ship) and the performance requirements associated with these functions.

The major output of the study is a set of recommendations concerning the future automation of SCC functions. Particular emphasis is placed on the integration of individual control actions into automated procedures and the integration of discrete items of monitored data into informative displays. The perspective of the analysis is both technological and psychological. That is, any manned-system design must be both technologically feasible and based on sound psychological principles.

2. INTRODUCTION

Modern technology, if applied in a manner sensitive to the human operator or decision maker, offers enormous opportunities for the provision of timely, accurate and integrated information, or speeding up task performance and for increasing operational effectiveness. The same technology, applied in an insensitive manner, provides equal opportunities for increased human error (and increased dangers consequent on such errors). Tasks can be designed that simply cannot be performed, or require performance that cannot be achieved with consequential waste of resources on a large scale.

---

* Dr Pechey was a Senior Consultant with Electronic Facilities Design Limited when the study was conducted.

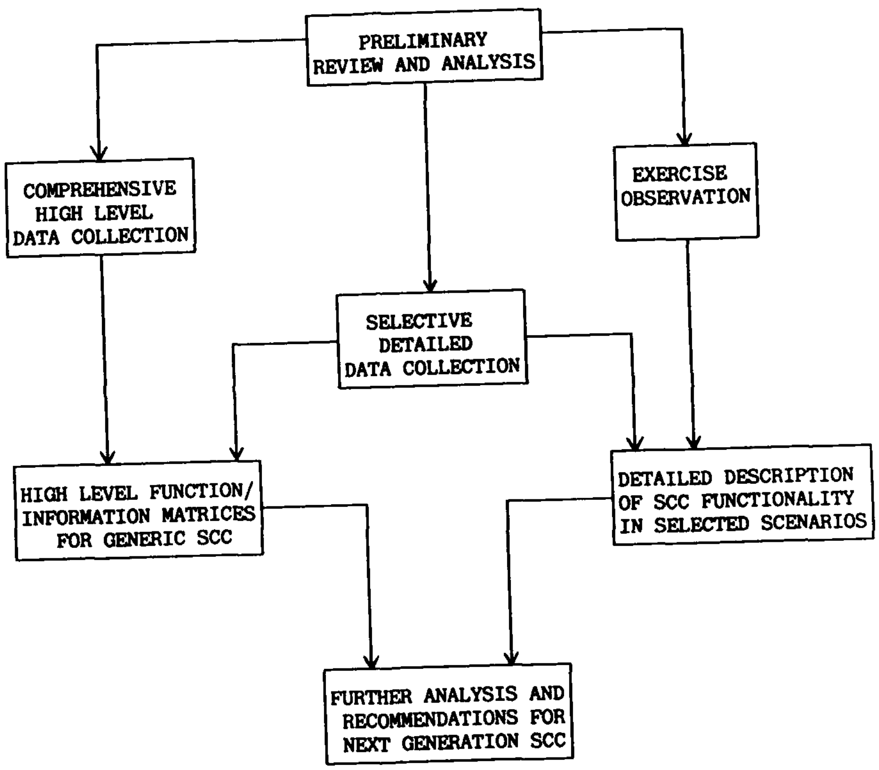


Figure 1 Study Methodology

MOD(PE) Sea Systems Controllerate is therefore engaged in a programme of forward looking studies, taking a user orientated functional perspective, in preparation for the procurement of the next generation of Ship Control Centre (SCC). A key component of this programme is the study described in this paper, which was jointly conducted by MOD(PE) Sea Systems Controllerate and Electronic Facilities Design Limited (EFD).

The main purpose of the study was to undertake a functional and performance analysis of SCC operation and on the basis of this analysis to provide recommendations covering possible approaches to the future automation of SCC functions. The study was not concerned with the SCC of a specific class of ship. Rather the focus of the study was a 'generic' SCC defined in terms of the major functions common to the SCCs of most classes of naval vessel.

This paper sets out to highlight certain of the more interesting general points arising from the study. The paper describes the study methodology, the functional and performance analyses, and outlines the general recommendations for future automation.

3. STUDY METHODOLOGY

3.1 Overview of Study Methodology

The data collection methodology was driven by the need to collect as much information as possible in a short space of time. It was also essential that the information be collected in a highly structured manner to aid integration and organised presentation, prior to further analysis. Given these constraints it was decided to carry out a comprehensive investigation of all SCC functions at a high level only and to support this high level analysis with a detailed investigation of selected areas of SCC functionality. An overview of the study methodology is shown in Figure 1.

Following an initial period of document review, informal discussion and preliminary analysis, data was collected by three complementary types of investigation, as follows:

a. Comprehensive high level data collection. Extensive discussions were carried out with a few highly experienced SCC personnel focusing on completion of comprehensive function/information matrices. This process is described in section 3.2 below.

b. Exercise observation. Observation of machinery control and Nuclear, Biological and Chemical Defence (NBCD) exercises was carried out by experienced human factors analysts.

c. Selective detailed data collection. A programme of structured interviews was conducted with key naval personnel focusing in detail on the tasks and functions performed by the SCC in selected scenarios. This programme is described in section 3.3 below.

These investigations allowed the collection of two different, complementary sets of data. Firstly, a set of comprehensive, but high level functional/information matrices. Secondly, a detailed description of SCC functionality in selected scenarios. Together these provided the necessary baseline data for further analysis and ultimately for making recommendations concerning the future SCC.

3.2 Comprehensive High Level Data Collection

The method for the high level data collection was particularly designed to elicit information concerning the flow of information associated with each task and any associated performance requirements. The method was based on elements of EFD's in-house requirements definition methodology, COMREFORM (COmbined Methods REquirements FORMulation). These elements are task and data flow analysis, and particular techniques for representing the functionality of a system.

On the basis of the initial document review and preliminary analysis, two skeleton function/information matrices were produced.

The Function/Data matrix is a matrix of SCC functions/tasks against data flow (and processing) requirements. The matrix (when completed) is designed to show, for each SCC task, the sources or destinations of information associated with that task, the typical current modes of information transfer and the role of the SCC with respect to the information.

The Function/Performance matrix is a matrix of SCC functions/tasks against performance and manning requirements. The matrix (when completed) is designed to show, for each SCC task, the associated accuracy, frequency, priority and response time requirements for different NBCD states and to identify the associated operator and supervisor in each state.

The task of completing the matrices was used as the basis for extensive informal discussions with selected experienced users. The matrices were completed in consultation with users who had experience of current in-service ships and who had been involved in various projects concerning the Type 23 Frigate. The experiences were drawn together during the interviews in which the matrices were completed, in order that a broadly common view of activities in a generic SCC could be achieved.

3.3 Selective Detailed Data Collection

The aim of the interview programme was to elicit detailed information concerning the functions and tasks performed in the SCC in selected scenarios. The interviews concentrated on a representative sample of the tasks which had been discussed at a high level in the context of the function/information matrices. The interview programme thus enabled two objectives to be met, as follows:

a. The interviews served to validate the information gained during discussion of the matrices.

b. The interviews enabled an expanded discussion of the selected tasks. Thus the tasks could be placed in an appropriate context, and the inter-relationship between tasks could be explored in greater depth.

The method used for the interview programme was EFD's Whole Process Structured Interview Methodology. This particular method was used because it enabled the collection of the maximum amount of information from interviewees in the minimum time. It ensured that information was collected from interviewees in a structured manner, whilst also allowing issues to be raised independently by the interviewee.

Each interview was based around discussion of incidents occurring in the context of two scenarios, as follows:

a. The first scenario exercised various aspects of machinery control. The interviewee was first asked to identify typical middle watch tasks that would be performed by SCC staff in a ship which was steaming in NBCD State 3. A series of incidents were then introduced into the discussion; for example, single generator failure, and loss of pitch control.

b. The second scenario exercised various aspects of damage control and was based on a ship receiving bomb damage from an aerial attack whilst in NBCD State 1. Damage incidents were then introduced into the discussion. These represented gradually increasing levels of severity, from the effect of the initial blast effect on alarms in the SCC through to major free flooding in two machinery spaces. This scenario being adapted from Flag Officer Sea Training (FOST) training exercises.

For each of the incidents included in the above scenarios the following issues were discussed:

a. How would information concerning the incident be received in the SCC, and who would receive the information?

b. How, and by whom, would that information be processed in the SCC? In particular, discussion was focused upon the degree to which specialist knowledge and experience was brought to bear on decision making.

c. What resulting information would be transmitted from the SCC? In what form and by whom?

Views were also sought as to whether tasks could, in future generations of SCC, be more efficiently executed, for instance by re-organisation of the SCC or increased automation of task components.

The output from the interview programme thus provided a detailed characterisation of SCC functionality in selected areas. This information,

together with that from the comprehensive high level analysis, provided the basis of the functional and performance analyses which are discussed further in the following section.

4. FUNCTIONAL AND PERFORMANCE ANALYSIS

4.1 Functional Analysis

4.1.1 SCC Functions. At the highest level the SCC has two primary functions, as follows:

a. Monitoring and control of machinery which enables the ship to fight, move or float.

b. Detection, assessment, containment and repair of damage which is sustained during fighting, moving or floating.

SCC tasks in support of these functions can be categorised in terms of the information flow involved, into six major categories. Monitoring tasks, control tasks, assessment tasks, reporting tasks, recording tasks and display tasks. The detailed results of the functional analyst were presented in the form of six data tables - one for each task category. Each table presents for each task falling into the category, various data including the source of input information to the SCC, the degree of task complexity and the typical degree of current task automation (the exact data presented varies according to the task category).

4.1.2 Monitoring Tasks. In order to perform the roles of machinery and damage control the SCC needs to obtain extensive information from all areas the ship. For machinery control purposes this information concerns primarily the state of machinery or systems such as temperature, pressure, tank levels or speed. For damage control purposes this information is more concerned with the detection of fire, flood or damage and with the activities of repair parties.

There are distinct differences between machinery and damage control monitoring tasks. For machinery control the majority of monitoring is carried out for detection purposes. There are also some feedback tasks concerned mainly with manoeuvring control and support functions. For damage control purposes, however, a much higher percentage of tasks are carried out to provide feedback or general information. This is because once damage, fire or flood has been detected the primary role of the SCC is to co-ordinate local control actions. In order to act as the co-ordinating agency the SCC must obtain information to guide damage control activities and situation assessment.

Information is obtained by the SCC either automatically via sensors and data links to the panel, alarm or warning systems in the SCC, or by verbal communication by SCC roundsmen or other personnel. Tasks related to the

machinery control role tend to be more automated than those related to damage control - in the sense that there are more sensors and data links to the SCC.

4.1.3 Control Tasks. These tasks involve both direct control of machinery from the SCC and the co-ordination of local machinery control or damage control activities. Control can be considered to be the primary SCC tasks. This is because many other categories of task are required simply for the purpose of providing information as to whether control tasks are required and subsequently to guide these control tasks. Control tasks relating specifically to machinery are more likely to be automated (that is, rely on data links) than control tasks relating to damage containment or repair. This is probably a consequence of the relative predictability of machinery control tasks. Considerable flexibility is required in the execution of damage control and this is currently achieved by using highly flexible human resources. Damage control tasks therefore typically involve local manual activities co-ordinated by the SCC by means of verbal communications.

4.1.4 Assessment Tasks. Assessment tasks for both machinery and damage control purposes typically involve assessing information collected from various sources and the application of detailed knowledge of the system, ship, NBCD procedures for decision making purposes. Assessment tasks include such activities as diagnosis, determining the need for action and carrying out calculations. The output of assessment tasks is typically either information relevant to the successful performance of control tasks or a decision that a control task should be initiated. There are currently virtually no automated decision aids to help SCC personnel with their assessment tasks.

4.1.5 Reporting Tasks. Reporting tasks involve providing the Command with necessary information relating to machinery or damage control. In the context of machinery control this information relates to the state and availability of systems pertinent to manoeuvring or to failures or incipient failures of machinery. In the context of damage control this information relates, for example, to repair activities or the NBCD environment.

Although SCC information sources are varied (including the SCC panel, rounds and verbal communications), reports are almost invariably conveyed to the Command by verbal communications. Reporting may be routine, and may also depend on detailed knowledge and experience, and on the awareness of when a situation should be reported and what information is of significance.

4.1.6 Recording Tasks. Recording tasks are required as the SCC needs to maintain medium and long-term records. Certain information relating to machinery control is automatically recorded. This typically includes machinery alarms and warnings and certain performance parameters. The remaining information is typically recorded manually on paper (usually in a log).

4.1.7 Display Tasks. Display tasks are almost exclusive to the damage control role. These tasks are necessary as the SCC must maintain a visible record of current information to co-ordinate damage control activities. All such damage control display tasks currently involve manual recording on the SCC incident board of information by verbal communications. This information includes the nature, extent and location of damage; the extent and location of flood or fire; the state of doors, hatches etc; and the state of local repair or control activity.

4.1.8 Overview. The vast majority of SCC tasks were classified as either monitoring or control tasks. Three monitoring-control task combinations are clearly evident in the current performance of the majority of SCC functions. These task combinations (listed in order of increasing level of automation) are as follows:

a. Receiving verbal reports from roundsmen or other personnel and co-ordinating men to perform local control actions.

b. Obtaining information from SCC monitors and co-ordinating men to perform actions locally.

c. Obtaining information from SCC monitors and carrying out direct control actions from the SCC.

There are currently few examples of the loop between automated monitoring and control being closed except by the man. Hence, a man is almost always required to respond to incoming information by selecting an appropriate action. That is, to make a decision as to when action is required and what action is required. Thus although only a small proportion of SCC tasks were classified as assessment tasks, there are many implicit assessment tasks related to the decision (on the basis of system monitoring) that control is required and relating to the exact nature of the control action required.

4.2 Performance Analysis

The main data set on which the characterisation of SCC performance requirements was based was the Function/Performance matrix. Aspects of the data were further discussed in the interview programme in order that issues such as performance priority hierarchies between tasks could be explored in greater depth. A high degree of performance is required of the SCC. The majority of tasks, when considered in isolation, were identified to be of high priority and requiring a high degree of accuracy. This is particularly the case for those tasks directly related to command priorities.

The SCC is involved in achieving all three Command priorities. That is, from providing an infrastructure to enable the ship to fight, move and float. When there is competition for limited resources, SCC performance priorities

are determined by command priorities. The current command priority will dictate the priority and hence the relative performance requirements for different SCC tasks.

In the case of reversionary action, although the performance requirement for fast and accurate response is maintained, it is accepted that absolute speed and accuracy of response will be reduced in comparison with normal function.

5. AUTOMATION IN THE NEXT GENERATION OF SCC

5.1 Overview

The study investigates from both psychological and technological perspectives, different options for the automation of SCC functions in the next generation of SCC. The discussion focuses on the need to attain a safe, workable and efficient man-machine system utilising to best effect the strengths and weaknesses of each contributor to the SCC functionality.

Control could theoretically be entirely automated (taking the man completely out of the monitor-control loop) by using, for instance, IKBS techniques. Given the current state of technology, however, it was considered that such techniques would be better utilised to assist the man in his decision making process than to replace the man entirely. The discussion was therefore primarily concerned with further provision for remote control and monitoring from the SCC and with automation of the SCC MMI itself. An overview of the main points of discussion in these areas is provided in the sub-sections below.

5.2 Provision of Remote Control and Monitoring from the SCC

There is considerable scope for further provision of links to enable system monitoring and control to be performed at remote SCC locations. (This is particularly true for machinery control tasks.) Digital techniques are recommended for component automation because of the increased flexibility this would allow for both monitoring and control.

The nature and extent of remote monitoring facilities should be closely guided by the informational requirements for feedback or decision making. For instance, sensors should directly measure relevant parameters. (If information concerning water flow is required, flow should be directly measured if possible and not inferred from pressure measurements). Unless operators receive both accurate and relevant information and have confidence in this information, the possibilities for increased efficiency and reduced manning will not be realised.

The provision of certain types of information to the SCC will be particularly difficult to automate. This applies to certain information collected by roundsmen but more particularly to the monitoring of damage control activities in State 1. Similarly the unpredictability of the control

activities required for damage control and the consequent flexibility required in task execution means that these control tasks involved in damage repair and containment will also be difficult to automate effectively. Damage control tasks in State 1 are therefore likely to remain relatively man-intensive. The flexibility of the human mind in making decisions in new circumstances and the flexibility of the human body for the execution of non standard actions provide particularly powerful tools in the damage control situation.

The provision for local entry of information into a data network, combined with automated displays and decision aids would nevertheless have significant impact on the current possibilities for the loss, distortion or misuse of verbally relayed information.

5.3 SSC Control MMI

RN ships existing method of control largely comprises the control of individual components, that is, one initiator (switch or button) in the SCC corresponds to one component (for example a fuel valve). This represents the simplest level of remote control of ship systems. The control of components can, however, be configured in a variety of ways to build automated procedures at increasing levels of complexity. There is a limited degree of such automated procedural control available on in-service RN ships.

Although step by step control of individual equipments from within the SCC gives a high flexibility of control, the approach is slow and vulnerable to operator error under stress. Where standard sequences of control steps (procedures) can be defined, it is recommended that these be configured into automated procedures.

Automated procedures can either be operator initiated or automatically initiated given the relevant conditions. Operator initiation of automated procedures is recommended in cases where the pre-conditions necessary for task initiation cannot be easily specified in advance (for example, where the nature of the response required is highly context dependent). This approach is very flexible although it does have time penalties.

Totally automatic initiation of automated procedures (that is, without any operator involvement) is recommended only in cases where the required response to an event is always the same and can be achieved safely. Response time is reduced but at the cost of reduced flexibility and a possible reduction in the operator's feeling of control over the functions for which the SCC is responsible.

The problem of the operator's perception of control can be overcome by provision of facilities for the operator to override an automatically initiated procedure. However, the requirement to make an override decision in a given (probably short) time may actually increase an operator's stress in situations where there are many issues demanding his attention. It is

recommended therefore that such procedures be confined to situations when time lag between initiation and action is long enough to allow unstressed decision making.

Given a well designed SCC MMI there is potential for the SCC workload to be significantly reduced by the adoption of extensive procedural control without compromising safety or efficiency.

5.4 SCC Display MMI

In the Type 23 there has been a considerable extension of automatic health and status monitoring and recording. There remains scope for further such automation in the future. In State 3 this will have a major impact on the workload of the roundsman. It is not recommended, however, that the physical observation of machinery is completely replaced by automated sensing. The breadth of the roundsman's senses combined with his knowledge of the machinery provides a powerful resource for detecting problems and incipient problems.

The impact of increased monitoring facilities on SCC workloads is dependent upon the level of SSC MMI automation. A major increase in the provision of raw, unfiltered and unintegrated data to SSC personnel is likely to increase the workload, possibly overloading them in State 1. One of the reasons why the typical 3-level hierarchy of SCC personnel is required in State 1, is to provide intelligent integration and filtering of raw information on its way to the decision maker. Digital collection of sensory data will create greater flexibility to integrate and display information in a number of forms. The requirement must be that the data are displayed in ways which reduce the mental processing that must be performed by the decision maker. The load on the SCC decision maker will depend significantly on the extent to which any information is intelligently filtered and presented.

5.5 Automated Decision Aids

With increasing use of automated control procedures which are man initiated or have provision for the man to override, the human activities associated with control tasks will become primarily decision making or assessment activities.

The provision of intelligently filtered and integrated information will clearly be a major aid to effective decision making in the SCC. There is also considerable scope for enhancement of permanent aides memoire (such as mimic diagrams) with dynamic displays and for the automated display of aides memoire which are needed only occasionally (such as underground maps) and of action check lists (for instance, representing default automatic control sequences).

In addition to the likely contribution of IKBS techniques to the intelligent filtering and integration of information, this technique could be

used in many ways to aid the decision maker. Potential contributions could include the following:

a. The provision of "what if" facilities. That is, facilities which allow SCC personnel to evaluate the probable effects of actions against a simulation of relevant system operation.

b. The provision of "expert" diagnostic aids for maintenance and damage/incident control (condition based monitoring).

c. The display of proposed corrective action in the case of incidents.

d. The provision of warnings of potentially hazardous consequencies of proposed operator actions.

e. The development of an "intelligent MMI" which would be capable of distinguishing between the "novice" and the "expert" and so provide the appropriate level of information.

6. SUMMARY

There is potential for significant increases in the extent of remote monitoring and control carried out from the SCC in the next generation of SCC. If human aspects are not fully taken into account, however, there is a danger that SCC operators may become overwhelmed with conflicting demands. Three factors will be of critical importance:

Firstly, the well designed integration of simple control actions into automated procedures can significantly reduce the workload but care will have to be taken to ensure that the man still ultimately retains control of SCC functionality and perceives that he has such control.

Secondly, the intelligent filtering and presentation of information and the provision of automated decision aids will also be essential if the full potential benefits of increased automation are to be realised.

Finally, safety procedures, which have been built up over many years of practical experience, will need to be critically re-examined in the context of the automated SCC.

Whilst the implications of not taking these issues into account may be serious, it is anticipated that if given due attention during all phases of the procurement cycle, adequate solutions can be found. Thus an automated SCC can be designed which need not degrade the high standards of safety and performance required in SCC functionality.

7. DISCLAIMER

The views expressed in this paper are those of the authors and do not necessarily represent the official views of the Ministry of Defence.

MARITIME MANEUVERING PILOTING AID

by Carmine G. Biancardi
Istituto Universitario Navale, Naples, Italy,
Massimo Capecchi
Istituto Tecnico Nautico, Rome, Italy,
Antonio Troiano
Istituto Tecnico Nautico, Naples, Italy,
Angelo Trotta
Istituto Tecnico Nautico, Procida, Italy,
Mario Vultaggio
Istituto Universitario Navale, Naples, Italy.

1. ABSTRACT

In this paper is presented an application of a knowledge based system for safe piloting of ships in navigation. The onboard system is useful when approaching narrow waterway, close collision situations, ports etc. improving the safety of ships and navigation. The objective of the system is to catch the experience and the heuristic of experienced seagoing personnel, engineering knowledge of ships and scenario and international laws. Using the system, an unskilled operator is able to perform the same kind and amount of work of a very experienced one. The system is also a tutoring and training tool. It is considered a DECISION SUPPORT SYSTEM (DSS) because it helps the operator and in the same time it improves the operator skills. Validation with real piloting trials shows how the system works in congested waterways and restricted water. The research guideline and some results are presented in this paper.

2. INTRODUCTION

Since the end of last decade, maritime industry is in a rapid structural reorganization. It is introducing management and operational techniques based upon the widespread use of computers and the "new information technology", in many cases demanding fundamental and sometimes painful adjustments to traditional practices and attitudes.

So far, modern merchant ships are highly automated and have been made so reliable that very little skill is required by those in control of them, whereas on the other hand, it is true that modern

merchant ships are highly complicated and therefore only those who fully comprehend their complexities should be trusted with their operation, [1,2].

A mariner's understanding of a particular vessel's operating characteristics develops gradually as experience with the ship accustoms the mariner to its handling qualities. Simulators and Advisory Systems are aids that can shorten the time needed to develop that understanding, [1].

Moreover, another point should be considered. For many years the number of people employed at any time on board a typical ship has been declining. In some types of vessels there has been a two, three, even fourfold reduction, brought about initially by the introduction of fairly crude mechanical automation and subsequently extended by the adoption of new technology. Regardless of how this process may be justified, it seems that is unlikely to be reversed easily. Thus we may expect to find a large proportion of the world's merchant ships operating with very few people on board before long. How then, can we ensure that all the skills which may be needed will be present in such a small groups.

Since the ship management is become a complex system monitoring and controlling is become higher complex, the ship-officers need a long training time and sea-experience to develop the expertise to master the problem facing their daily task.

There is now evidence to suggest that certain marine management tasks can be performed more efficiently by machine than by a ship's officer. The emerging science of Artificial Intelligence (AI) has made possible the development of computer programs used in monitoring, controlling and simulating the behavior of fully automated vessels, [1-8]. The advent of such programs means that, in the future, ships will be operated by a combination of human and machine intelligence.

In September 1987 a project was initiated at Department of Navigation of The Istituto Universitario Navale at Naples, Italy, to design and develop a research prototype decision support system for the maritime pilotage, [6,9]. The system is intended as a rehearsal tool for masters, mates on watch and pilots on board merchant vessels transiting congested waterways but can also be used as a training tool for junior pilots and deck officers. It employs the captured decision-making expertise of experienced masters and scientists as well as environmental information to provide the ship operator with piloting recommendations. The focus of the project was to provide a useful tool for ship-officers and pilots that would automate data management tasks while providing computational capabilities to enhance the "pilot"'s decision-making process. This paper provides detailed account of the design

and development of this piloting tool. It first defines the piloting complexity and then presents a summary of the Knowledge base, the system configuration and validation. The system is then discussed regarding its architecture and modules.

3. PILOTING COMPLEXITY

The term "piloting" has been concerned only for navigating in harbors, canals and congested waterways. In this study the word "piloting" means the "drive" of the ship, [6,9]. So, whenever a ship is in navigation, she is in a "piloting situation". Such a definition causes a wider generalization of the piloting problems.

Ship-officers and pilots have to face a daily duty with several constraints and information such as national and international rules, meteorological impact on the ship, ship maneuvering behavior and other data.

A symbolic description makes use of something that stands for something else by reason of relationship, association, convention or accidental resemblance. For example, the lion is often used as a symbol of courage. In order to develop a symbolic software it is necessary to resort to an heuristic approach. Heuristic approach is not an exact mathematical procedure but it is useful to obtain results, that after validation, can describe with accuracy the real world. The heuristic operation and its omonymous approach consist in collecting and elaborating documents, data, information, expertise etc that can bring the discover or clarification of a fact, a behavior sequence, etc. Usually the heuristic approach is typic of history science but lately it is been successfully used in very different fields such as engineering [1]. The applicability is best illustrated by example. If the ship is in navigation the number of the parameters which can be monitored is fairly large. The number of parameters which are immediate control of ship-officer is somewhat 100 or more. However, in one way or another, each of the monitored parameters effects the operating efficiency and safety. This implies an infinite number of choices. From a practical point of view, however, there are fewer parameters which must be controlled and therefore fewer combinations of legal values of the state variables. It is nevertheless still a very large problem to solve by a strict mathematical approach.

The Maritime Maneuvering Piloting Aid (MMPA) is intended to decrease the information overload under which the pilot presently labors, thus providing better piloting and increasing the safety of navigation for vessels in a close waters situations; to provide for more effective distribution of local piloting knowledge within individual pilot organizations, thus providing more efficient and consistent knowledge transfer; to serve as a training device

onboard merchant vessels, pilot boats, and ashore ship simulators for the training of junior pilots and deck officers, [6,9].

The task of this system is to bring the ship on the "best" path with the lowest degree of risk among the infinite available paths.

4. SYSTEM CONFIGURATION

The Maritime Maneuvering Piloting Aid is a computer program designed to simulate the expertise of a human pilot specialist. Its primary components are a knowledge base and an inference mechanism. The system knowledge is composed of facts and heuristics. Factual knowledge is information known to all experts in the piloting prescribed field and is considered exact. It is sometimes refered to as "book" knowledge. Heuristic knowledge is considered inexact. It is made up of private, even unconscious rules of thumb that characterize the expert decision making. The MMPA can be also referred to as knowledge system or intelligent assistant. Programs of this type are equipped with a limited number of rules (a few hundred) and are intended to play a supporting role in a highly constrained setting. Developing a small PC-based expert system shell, the research prototype program has been realized on a personal computer. It has been realized using symbolic and algorithmic programming techniques. The prototype has been developed to provide decision support to masters, mates on watch and pilots aboard merchant vessels in congested waterways.

The inherent complexity of the system-building process and the fact that the process was not completely known at the beginning precluded laying out many of the steps in advance. As a result, we have found than an evolutionary prototyping strategy, proceeding from simple prototypes to more complex models, proved most effective. So, an iterative approach has been used to develop such system.

In an overall look of this study, the following three main steps were followed in the development of this program:

a. acquisition of knowledge base
b. system architecture
c. validation of the research prototype

4.1 acquisition of Knowledge base

The knowledge base is briefly outlined in fig. 1. It consists of different modules that can be itself considered as independent knowledge base. In this way, a knowledge book has been built, it consists of rules of road, ship stability calculation, ship maneuvering simulation, ship weather behavior, navigation rules,

rudder and engine maneuvering rules, cartography, bathymetric navigation, emergency rules. Moreover, an experiment was conducted for collecting the protocols of pilots, master mariners and mates, by mean of special interviews. The aim of the protocol analysis was twofold:

a. to insure a systematic means of identifying, codifying, tracking, and incorporating heuristics into the prototype advisory system that initially contained only "book knowledge,"

b. to use the experience derived from the experiment to develop guidelines for knowledge acquisition for other researcher and practioners.

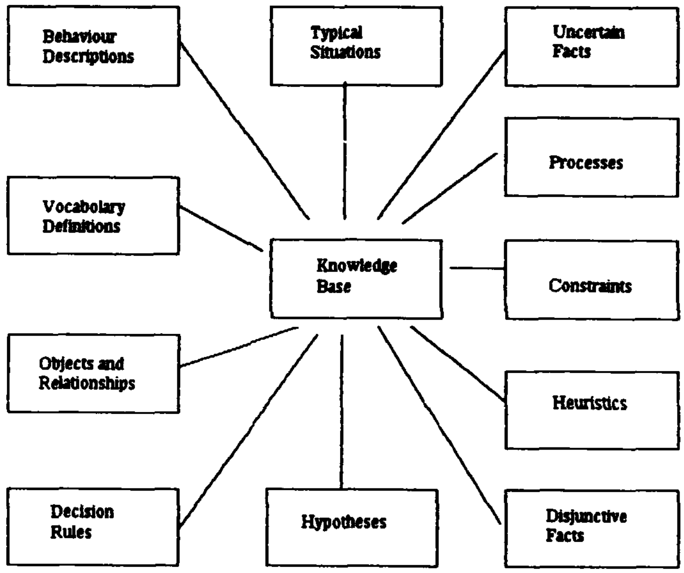


Figure 1. Diversity of knowledge which has been incorporated into the MMPA.

4.2 System architecture

The system architecture is a semantic net. Fig. 2 shows the semantic net for the onboard advisory system. The principal features of this conceptual design are:

a) the integration of multiple, semi-indipendent modules, each containing knowledge about a separate sub-problem.

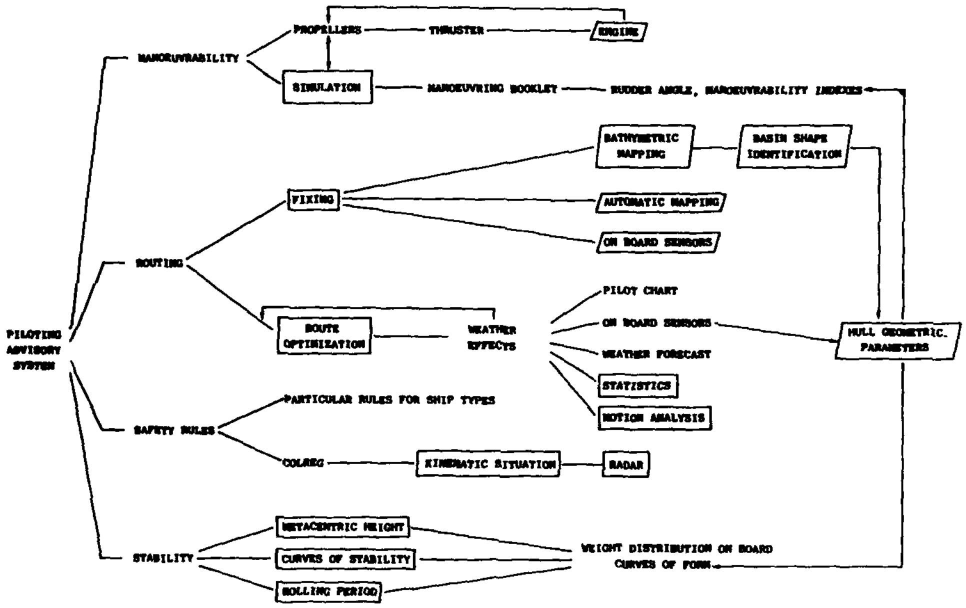


Figure 2. Semantic Net

b) the existence of a knowledgeable controlling module which integrates various other modules of the system. The other modules are either knowledge-based systems themselves or algorithmic programs for processing data.

In fig. 2, the module identifiers which are include in boxes are algorithmic routines that are written in a conventional programming language. The other identifiers are modules which are appropriate to construct as a rule-based system.

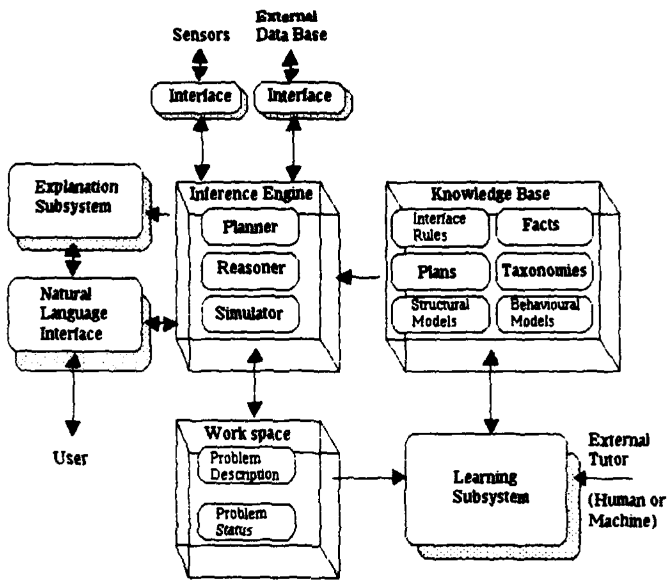


Figure 3. MMPA System Components

a. MMPA system components. Three main parts constitute the system as it can be seen in fig. 3:

a.a. Knowledge Base. This parts contains the heuristics of the problem (inference rules and facts), expertise and the experimental and numerical data-bank (facts). Moreover, it has the possibility "to learn" from the expertise developed from the daily working use.

a.b. Reasoner Filter. It is a classic "reasoning filter" such as: IF.....THEN; that make a confrontation between inference rules and facts and choose the most appropriate numerical method and solution.

a.c. Natural Language Interface. The Language Interface gives to the operator an opportunity to actively be involved and to understand and prepare the problem.

4.3 Validation

The validation of the research prototype can be summarized in two parts: one is the validation of the numerical results with full

scale tests and the acceptance that rules, facts and human expertise is considered as true. The second part is the operational evaluation of the advisory system in working conditions. This latter evaluation associated with the system prototype development is whether masters or mates on duty in a piloting context will make better decisions as a results of the use of the system. Some pictures of an operative full scale tests of M/V Rondine (RONDI) are presented in fig. 4, fig. 5 e fig. 6. The preliminary test has been performed in the bay of Naples with the ship communicating with a VTS center as described in section 5.2.a.. The navigation consisted in leaving the port of Naples heading to the island of Capri. The bay of Naples is an high density traffic area. Some considerations can be outlined from the preliminary results:

Ship's path rappresentation on monitor of M/V "Rondine"

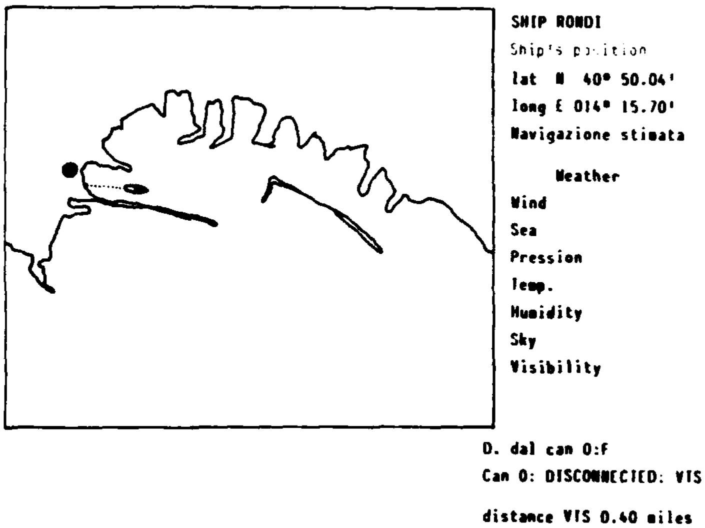


Figure 4.

a) use of the MMPA in a pilotage planning and training context seems to be a necessary consequence.

b) use of the system in on-line operation and its contribution to the decision-making equation is not yet completely understood. Supplementary research should be done within this area.

Ship's path rappresentation on monitor of M/V "Rondine"

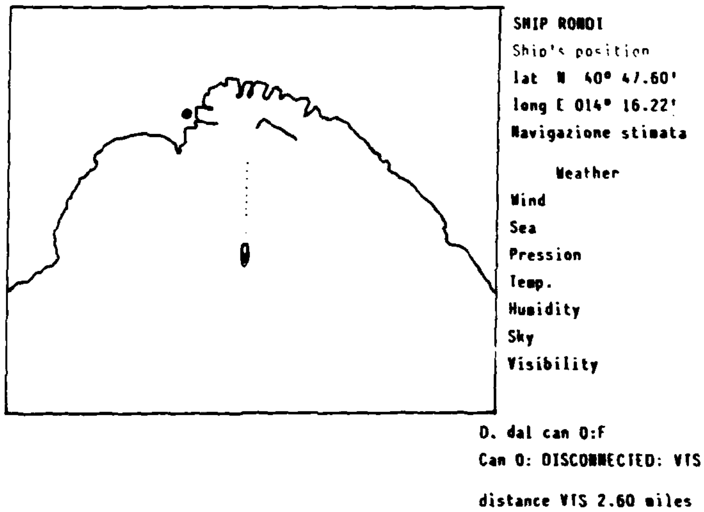


Figure 5.

Ship's path rappresentation on monitor of M/V "Rondine"

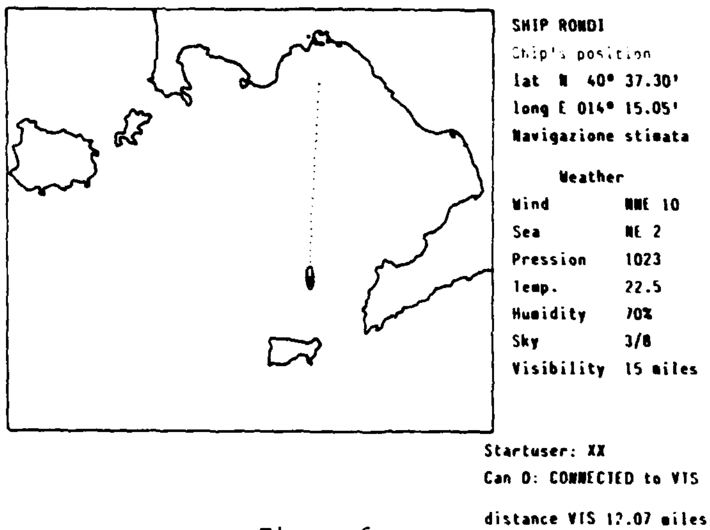


Figure 6.

5. MODULES

In the following chapter is briefly examined the modules' contents of the semantic net.

5.1 Maneuverability

The main features of this module can be summarized in three parts:

- on-board maneuvering simulator (fig. 7)
- mathematical rudder angle advisor
- propeller-rudder interaction effect (fig. 8 e fig. 9)

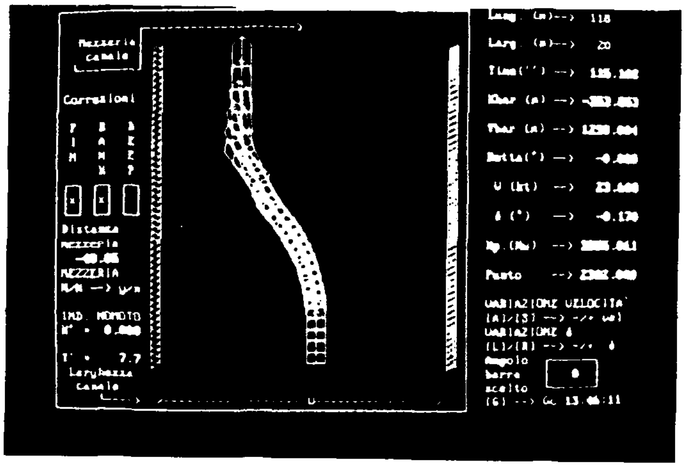


Figure 7. Onboard Maneuvering Simulator-advisor

The hybrid algorithmic-symbolic module employs a simplified mathematical model, [20], and a set of realistic expectations of ship behavior drawn from master's expertise and full-scale tests, to predict the results of imminent maneuvers. It includes hydrodynamic and powering characteristics of the ship, channel geometry, [14,17]. The module employs sensors for continuous updating of ship trim and environmental data.

SHIP'S STEERING

MENU

1 . Ship with single clockwise propeller
2 . Ship with single unclockwise propeller
3 . Ship with twin propellers
4 . ______________
5 . ______________
6 . ______________
7 . EXIT

STEERING OF A SINGLE PROPELLER SHIP

MENU

1 . Engine ahead: ship stopped or dead slow ahead
2 . Engine and ship move ahead
3 . Engine astern: ship stopped or dead slow astern
4 . Engine and ship move astern
5 . Ship ahead and engine astern
6 . Ship astern and engine ahead
7 . Thruster
8 . MAIN MENU

Figure 8.

MANOEUVRE

MENU

1 . Rudder in centre line
2 . Rudder to starboard
3 . Rudder to port

Solution for: Rudder to starboard

Bow begins to go to starboard and after tacks *slowly to port. Sometimes* the bow tacks to starboard.

Figure 9. To choose a maneuver means to make the bow turning as a function of rudder angle and ship speed. The output is not just always one as it may occur in some events.

The greatest technical difficult arose from the formulation of the abstract knowledge such as geometric shape of the waterway into concrete rules. It could be solved involving pattern recognition and image processing. However an easier solution has been found combining ship positioning with a suitable automatic bathymetric mapping of the navigation area, [18]. In this way a quick geometric shape recognition of the area is found, (fig. 10).

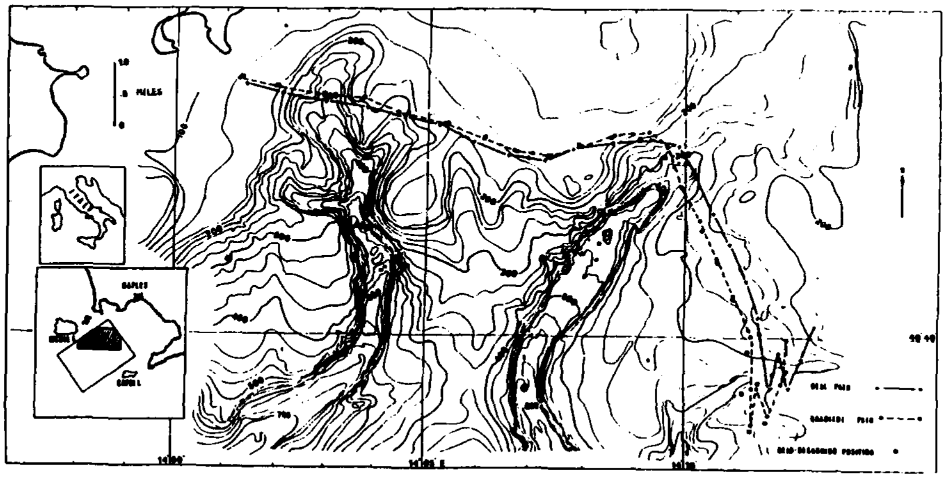


Figure 10. Bathymetric Navigation System for basin shape recognition.

The on-board simulator works either as a normal shiphandling simulator either as a supporting tool for the rudder angle display. The rudder angle advisor suggests the rudder angle as a function of the desired heading and a presumed time.

The propeller-rudder interaction effect examines the various combinations of the propeller/s and rudder/s. Moreover, it gives the expected ship behavior, [20], based on expertise drawn from experienced pilots, masters and mates, fig. 8 e fig. 9.

The decision making element is based mainly on observation of the predictive display, which only indicates the predicted heading after a fixed prediction time. The operator, using the course predictive display as a maneuvering aid, will only use the display information that accords with his criteria of satisfaction. In all other circumstances he will use his internal model to decide

rudder corrections.

5.2 Routing

a. Fixing. Using a priority filter, the module chooses the "best" radio-navigation system among the different available ones (LORAN, GPS, OMEGA, NNSS, etc.) then it makes a confrontation with the dead reckoning navigation. It uses a mathematical model to have an optimization of the ship fixing. The fixing is available on-line on the map, warning on the presence of navigation risks such as shoals and wrecks.

Moreover, it includes an automatic mapping system. In addition, by mean of an updated bathymetric navigation system, [18], this module gives information on the geometric shape recognition to the module described in section 5.1, (fig. 10).

Furthermore, the module can interact with VTS (Vessel Traffic System) centers by means of telematic messages, [16,24]. At the Istituto Universitario Navale has been developed a new concept of VTS that is not longer constrained by the use of radar but it employs a communication system. Onboard, the ship position and other data are coded in telematic messages and send to the ashore VTS center. The MMPA is VTS-ready. It automatically informs the VTS center of ship position and other information such as vessel traffic conditions, meteo-marine conditions, time of arrival and forward request for pilot and/or tugs [24].

b. Weather Effects. This module contains: updated meteorological information, extreme weather effects, wind-wave interaction effects, expertise in particular situations, I.M.O. (International Maritime Organization) meteorological criteria, optimization of ship path, [10-13]. Using this module, the operator can have on request: ship weather behavior, advices to solve realistic bad weather situations, advices and strategies to avoid bad weather.

A part of this module has the objective to select a route which minimizes the time and the cost and to optimize the ship safety taking into account actual weather effects. Input may be directed from external databases, however, this is probably not very practical for shipboard use. We assume that the required weather information is input by the user in some manageable form or there is some readily available database which is accessible by the advisory system. Output will be to a terminal video in the form of a recommended route.

In input the module requires meteo-marine elements such as wave direction and heights, wind force, etc.. The video output consists on information as rolling angle, motion resistance, others. In

addition, it shows the advices and the actions such as change of heading, speed variation, reduction of metacentric height and others. The bad-weather piloting flowchart is outlined in fig. 11.

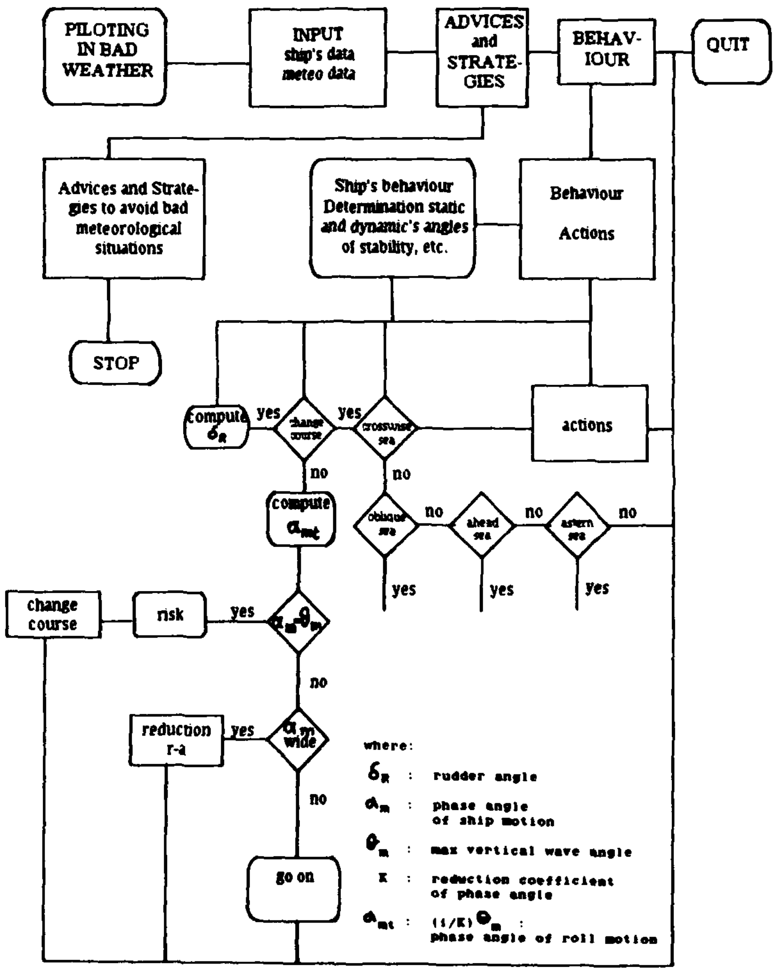


Figure 11. Piloting in Bad Weather.

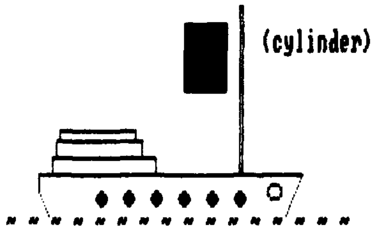
5.3 Safety rules

Taking into account the International rules of I.M.O. (International Maritime Organization) COLREG 72/81/89 as knowledge base, [21,22], the module, firstly identifies the situation and the target ship type and after gives the behavior rules for collision avoidance. Moreover, as a training tool it permits to know the definitions, visibility, positioning and technical details, shapes and sound signals required to be carried out by vessel types upon high seas.

```
                     (A)  -  Situations
Power-driven Vessels Under way      L>50m.   ...........(a)
  ..    ..    ..    ..   ..          L<50m.   ...........(b)
  ..    ..    ..    ..   ..          L<12m.   ...........(c)
Sailing Vessels Under way ...............................(d)
Fishing Vessels  ........................................(e)
  ..       ..     engaged in trawling  ..................(f)
Vessels not under command  ..............................(g)
Vessels restricted in their ability to manoeuvre .......(h)
Vessels constrained by their Draught  ..................(i)
Towing and Pushing ......................................(j)
  ..          ..     (when ship is unable)  .............(k)
Vessels engaged in dredging or underwater operations ...(l)
Vessels or object being towed  .........................(m)
Vessels engaged in minesweeping operations ............(n)
Pilot Vessels  ..........................................(o)
Anchored Vessels  .......................................(p)
Vessels Aground  ........................................(q)
Air cushion vehicle (in non-displacement mode)  ........(r)
                 What's situation ?
                 Table (A) permits to identify
                 a ship for every situation
```

Figure 12.

VESSEL CONSTRAINED BY HER DRAUGHT (rule 28)



by day

NEW IDENTIFICATION \ MAIN MENU \ SYSTEM (i/m/s)

Figure 13.

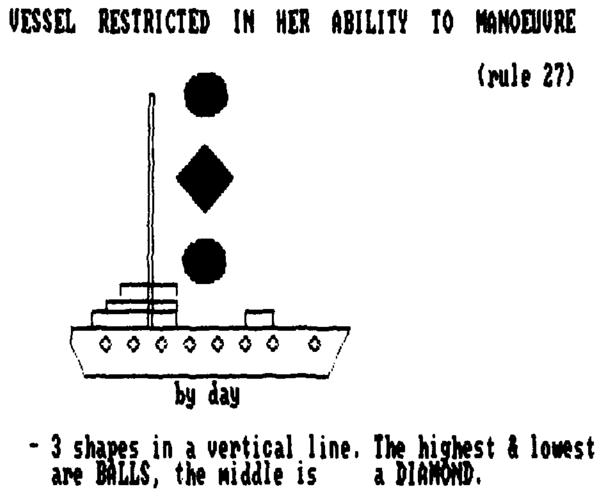


Figure 14.

Fig. 12 presents the video display menu to set up or to choice a situation. Fig. 13 e fig. 14 present a result of an identification process of two particular ships.

5.4 Stability

The module examines that the vessel meets the minimum static and dynamic stability requirements. This part furnishes advices and works as a support knowledge for the other part. This module uses for calculation the common formulae available in literature, [23]. Moreover, the module can work with external stability software package. In case of carriage of grain, it is considered the ship satisfying SOLAS requirements.

6. CONCLUSION

To catch the human expertise developed in several years of duty and the development of the expertise of knowledge engineers and to make it available to people with not enough expertise is the goal of the MMPA.

On board the system can be used as dual rehearsal tool:

- giving details in out-coming maneuver and the "best" path to reach with that speed and rudder angle and/or controlling other devices;

- training personnel during real-trim condition.

The development of a piloting expert system for training is a main contributions to the improvement of the seagoing personnel for the following motivations:

- using on regular basis the MMPA on board the deck-officer can acquire sensibility and tranquility both very precious when approaching dangerous maneuver;

- because the advisory system can evaluate the maneuvers, the operator can improve his skills by confrontation;

- the on-line use during a maneuver will be of a great assessment during any type of maneuver.

Ashore, as an intelligent tutor, it can contribute to the education of midshipmen in Nautical Colleges, to the licensing of ship's officers and the continuing education programs of both industry and academia, [1,9].

ACKNOWLEDGMENTS

Several people have been working on this project. We thank all of them. Particularly, we would like to thank Capt. Domenico Vacca and Capt. Francesco Ortino for their professional consultation. We are also grateful to Capt. Salah Benouarets for his professional suggestions and tips in preparing this paper. This research has been possible because of grants of the CNR (National Council of Research of Italy) and MPI (Ministry of Education of Italy).

REFERENCES

(1) C.G. Biancardi and D.R. Dellwo, "Machine Management of Marine Operations," Marine Technology, SNAME (Society of Naval Architect and Marine Engineers) U.S.A., Vol. 25., No. 1, January 1988.

(2) R. Balestrieri and C.G. Biancardi, "The Human Errors in Maneuvering of Ships (in Italian)," Studi Marittimi, Anno VIII, No.25, Napoli, Italy, Giugno 1985.

(3) S. Baron and C. Feehrer, An Analysis of the Application of AI to the Development of Intelligent Aids for Flight Crew

Tasks, Report prepared by Bolt Beranek and Newman, Inc., Cambridge, Massachusetts, USA, for NASA Washington, DC, October, report No.: NASA-CR-3944, contract/grant: NAS1-17355, 1985.

(4) A. Barr and E. Feigenbaum, "The Handbook of Artificial Intelligence,", William Kaufman, Inc., Los Altos, California, USA, 1981.

(5) J. T. Dillingham and A. N. Perakis, "Application of Artificial Intelligence in the Marine Industry: Problem Definition and Analysis, Final Report, Vol.II," Technical Report to Maritime Administration U.S. Department of Transportation, No. MA-RD-770-87016, 1987.

(6) C.G. Biancardi, M. Capecchi, A. Troiano, A. Trotta, M. Vultaggio, "The Piloting Advisory System," Proceedings of MARSIM & ICSM 90, Tokyo, Japan, 1990.

(7) M. Grabowski, "The Piloting Expert System," Fleet Management Technology Conference, Baltimore, Maryland, USA, May 5-6, 1987.

(8) M. Grabowski and W.A. Wallace, "Decision Support to Masters, Pilots and Mates on Watch at Sea and in Close Waters: an Application of Expert Systems Technology," Fleet Management Technology Conference, Boston, Massachusetts, USA, April 15-16, 1986.

(9) C.G. Biancardi, M. Capecchi M., G. Mallardo, A. Troiano, A. Trotta, "Expert System for Safe Piloting of a Ship (in Italian)," Annali della Facolta' di Scienze Nautiche, Istituto Universitario Navale, Vol. LVI, Napoli, Italy, 1988.

(10) R. Motte, Weather Routing of Ships, Maritime Press, London, U.K., 1972.

(11) S. Sannino, Meteorologia Nautica, Italibri, Napoli, Italy, 1983.

(12) E. Spinelli, "Calcoli dei movimenti e delle sollecitazioni di una nave in mare tempestoso per mezzo del metodo dello spettro di energia e della teoria della sovrapposizione," La Marina Italiana, No.1, Genova, Italy, 1964.

(13) U. F. Costaguta, Fondamenti di Idronautica, Hoepli, Milano, Italy, 1981.

(14) R. Balestrieri, C.G. Biancardi, S. Mauro, "Validation of a Simplified Mathematical Model for the Control of Ship Path (in Italian)" Tecnica Italiana, No.4, Trieste, Italy, 1989.

(15) R. Balestrieri and C.G. Biancardi, "The Maneuvering Simulator-Advisor", Marine and Offshore Computer Applications, Computational Mechanics Publications: Southampton, Springer-Verlag Berlin Heidelberg: New York, 1988.

(16) R. Balestrieri and C. G. Biancardi, "Hydrodynamic Parameter Consideration for Vessel Traffic Control in Congested and/or Restricted Waterways," Proceedings of VTS in the Mediterranean, Genova, Italy, 1990.

(17) R. Balestrieri R. and C.G. Biancardi, "Numerical Evaluation of Ship Maneuvering Performances in Confined and Shallow Water," PRADS'89, Varna, Bulgaria, 1989.

(18) V. Nastro, A. Russo, R. Santamaria, A. Sposito, M. Vultaggio, "Navigation by Gradient of Geophysical Parameters," Proceedings of the Fifth International Congress of the International Association of the Institutes of Navigation, Tokyo, Japan, 1985.

(19) Marine Research and Training Center, Shiphandling, Port Revel

(20) N. Petronzi N. and A. Vecchia Formisano, La manovra delle navi, A. Vingiani, Torre del Greco, Italy, 1984.

(21) Edward V. Lewis, Principles of Naval Architecture, S.N.A.M.E. (The Society of Naval Architecture and Marine Engineers, Jersey City, U.S.A., 1988.

(22) I.M.O. International Maritime Organization, International Regulations for Preventing Collisions at Sea, 1972 (COLREG 1972), Amendments adopted by resolution A.464 (XII) in November 1981, Amendments adopted in November 1989.

(23) G. Mannella G., Normative di Sicurezza Marittima, A.P.E. Mursia, Milan, Italy, 1984.

(24) C. Fusco, A. Sposito, M. Vultaggio, "Setting up a Local VTS Center Using VHF Computer-coded Messages," Proceedings of VTS in the Mediterranean, Genova, Italy, 1990.

AN OBJECT-ORIENTED DESIGN METHOD FOR SHIP AND MACHINERY SIMULATION.

by Keith Reading
Mary Ward
and David Bagge
Hawker Siddeley Dynamics Engineering

1 ABSTRACT

The requirements for simulation of ship systems are numerous and varied. This paper reports on the experience of producing a framework and toolset for implementing such simulations. It considers the characteristics required of a simulation algorithm language for many ship related applications, explains the solution chosen, and the conclusions derived from applying it. It also describes the authors experience in the use of multiple INMOS Transputers and the use of Object Oriented design and C++ coding in this context.

2 INTRODUCTION

In the course of our business as control system producers we frequently need to produce simulators. However these simulators are needed for a wide variety of purposes, and each purpose places differing emphasis on different aspects of the simulator. The purposes range from a purely internal investigation into the control algorithm to be used to regulate some piece of plant, through providing a test bed to exercise the final hardware implementation of a governor, to providing an operator training facility for a comprehensive machinery control and surveillance system. In the past these separate needs have been separately addressed, perhaps using a one-off computer program for the control algorithm investigation to simulate both the controller and the plant to be controlled, a digital/analogue hybrid computer and program to provide the real-time simulation for the physical governor test bed, and an individually developed set of hardware and software to produce the operator training simulator. With todays advances in digital computing hardware it was recognised that the same computational elements could be used to satisfy all of the above requirements. The challenge was seen as that of providing a framework of simulation system software to do the same.

3 OBJECTIVES

The principle objective was to produce a system of hardware and software modules which would allow us to satisfy the diverse requirements for simulators, serving a variety of end uses, in a quick and economical fashion.

To this end it was seen as necessary to identify, by analysis, the range of features required by such simulators and so arrive at a minimal subset of features which could be used to satisfy the range of known requirements. Experience told us however that requirements are not static and that the set of features satisfying todays known requirements would very quickly be found wanting in tomorrows application. It was therefore deemed essential that, whilst every effort be made to correctly identify the features necessary for the anticipated requirements, the set of features should be easily and cheaply extended or modified. It was further recognised that a key element in achieving this ease of extension or modification was to make the implementation of the separate features as independent and divorced from each other as possible. It was in this last area that the use of an object-oriented design and language seemed to offer more than previous techniques. It was also seen as highly desirable that the application algorithm writer should not need to strive for run-time efficiency. This was the reason for deciding that the simulation algorithm(s) should be executed by an easily extendible array of transputer processors.

In summary then the objectives were set out as follows:-

a) meet customers requirements for specific simulators

b) produce reusable simulation shell

c) evaluate use of object oriented language

d) extend experience of use of transputers

4 DESIGN

One of the earliest design decisions taken was that the simulation algorithm(s) should be written in a tailored but constrained "simulation language". This followed from the recognition that the algorithms would be the part which would be rewritten for each application and would be the area which would undergo continuous development during each project and consequently would be the area which would need to be debugged for each application. The design activity was thus split into the design and support of the "algorithm language" and the "system software".

4.1 Algorithm Language

Experience has shown us that 90% of the effort in implementing a simulation goes into obtaining the necessary data about that which is to be simulated. This inevitably leads to the coding and debugging of the algorithm needing to be done in a very short timescale. For this reason we had decided to employ an "algorithm language" which avoided the possibility of the writer making most of the usual programming errors and in particular avoided the chance of the type of error which is very difficult to locate and isolate.

To this end we defined a very simple language which had no flow control constructs and gave the writer no choice of data types and no pointer or indirect variables. Furthermore arithmetic operations were defined in this language in such a way that there was no such thing as arithmetic overflow.

It would have been nice if we could have also permitted the algorithm writer to be unaware of full scale and resolution limitations. In the case of input and output variables where such practicalities are an unavoidable feature of the hardware interface we felt that the algorithm writer had to be aware of these limitations. For intermediate variables used in the algorithm however we can remove such concerns by enforcing the use of floating point representation of adequate resolution.

The resulting language consists of

a) three groups of variables, External inputs (EIV), Internal (IV), and Output (OV)

b) an Expression table having one entry for each Internal and Output variable

c) a number of tables representing one and/or two input custom defined non-linear functions

The Input and Output variables are each subdivided into boolean and analogue types. The boolean type can only have the value true or false, whilst the analogue type is represented by a 12 bit signed integer. Internal variables are all held as floating point numbers (64 bit IEEE format).

The Expression table is an ordered list having one expression for each Internal and Output variable. An expression consists of zero to three terms. A term consists of a variable reference, a coefficient and an operator. The variable reference returns the floating point form of the present value of the variable which may be any one of the External input, Internal, or Output variables. In the case of boolean variables floating point zero is returned

for false and one for true.

The interpreter processes each expression in the table, in table order, and stores the result in the corresponding variable. Where the variable is of analogue type (Inputs and Outputs) the floating point result is rounded and limited to 12 bit integer. Where the variable is of boolean type then the value false is stored if the floating point result is zero, for any other result the value true is stored. For Internal variables the floating point result is stored.

The arithmetic operators supported are add, subtract, multiply, and divide. The operators returning boolean results are less than, greater or equal, logical inversion, and logical or (with the convention adopted, multiply is equivalent to logical and). The two nonlinear operators give access to linear interpolation in one and two input tables defined by the application programmer. For these operators the coefficient present in all other terms is replaced by the identifier of the table to be used.

It was also recognised that simulations in particular fields would give rise to the need for other operators or functions which it would be too inefficient to code directly in the simulation language. For example we might require trigonometric functions. The constraint was therefore placed on the design of the expression interpreter that it should easily permit the addition of further operators and functions.

At the initial design phase it was recognised that we would need some form of "algorithm language preprocessor". Although the expression table is printable ASCII in its loadable file form it is not programmer friendly. We planned therefore to generate a simple "algorithm assembler" to permit the programmer to create his code in a more comfortable format and permit a free choice of variable names.

4.2 System Hardware

As the system was designed to form the framework of a general, reusable simulator, it was decided that dedicated hardware should not be designed. The system should be built up from a number of available, separate components to meet the required simulator specification.

A PC was used as the basis of the system, functioning mainly as the operator interface since it provides keyboard, screen and disk facilities. It also provides a cheap hardware interface by use of the expansion bus into which a number of i/o boards can be slotted.

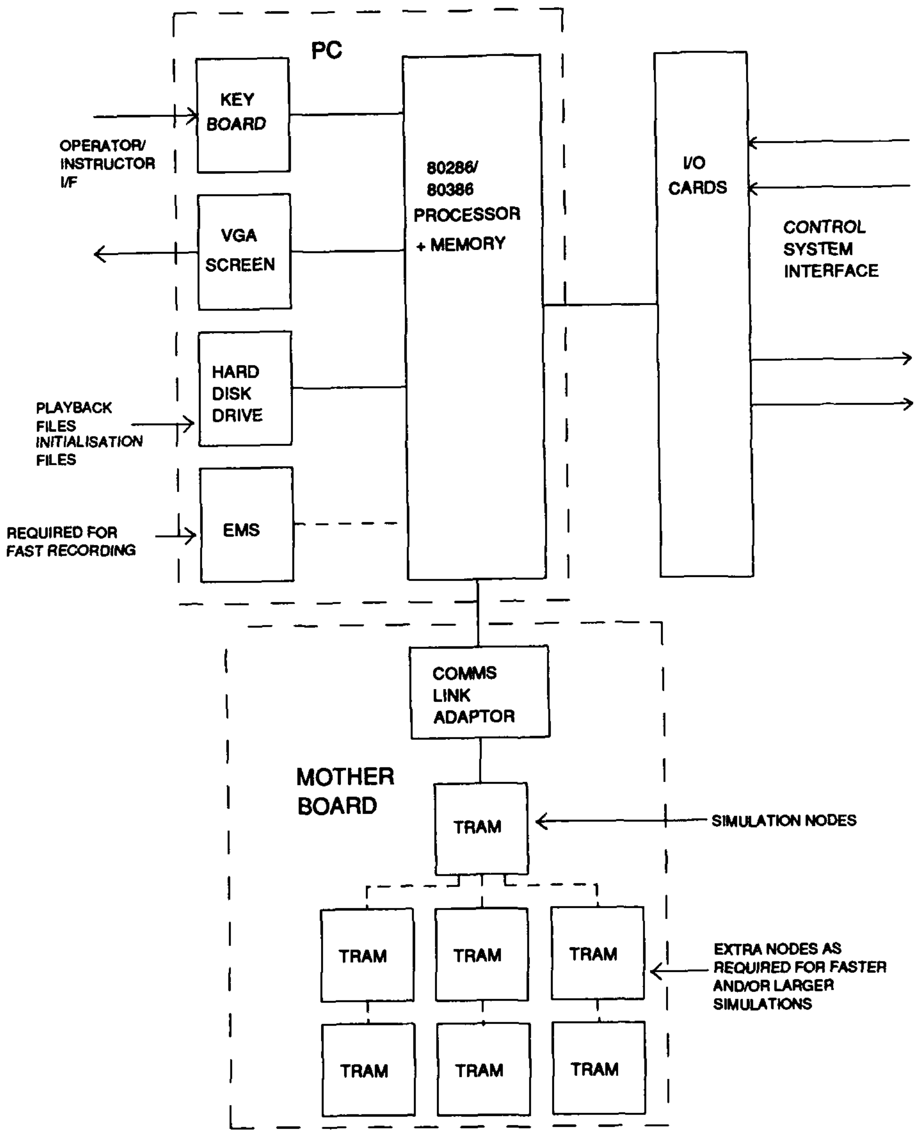


Figure 1 Simulator hardware

As the size of simulation algorithms and the frequency at which they are to be executed would vary, it was decided that these should not be PC resident. A network of modular processors would be desirable that could be built up according to the simulation size and performance required. To achieve this, it was decided to use transputer modules (TRAMs) equipped with 2Mb of RAM and a transputer. These suited our purpose well as they comprise only processor, memory, and the 4 INMOS links and come in various memory sizes. They are supported by a range of motherboards for various system buses including VME and IBM PC. This hardware configuration gives us the benefit of expandable execution power and memory since each transputer module is easily networked to others in flexible configurations by use of its 4 INMOS links. The overall hardware configuration is shown in figure 1.

4.3 Operator interface

The operators control of the simulation system is totally menu driven. The top line of his display always shows the current options whilst the second line gives an expanded explanation of the highlighted option or indicates the further choices that this option will lead to. Options are selected by moving the highlighted selection along with 'left' and 'right' arrow keys and then pressing return, or by typing the initial letter of an option. (option keywords are selected so that any individual menu does not contain two keywords starting with the same letter.) The menu system not only controls the operation of the simulator (e.g. Simulate, Freeze, Replay, Initialise etc) but is also used to choose which of the predefined views or pages is shown on the rest of the operators display. These pages are of three basic types:-

a) Tabular status display pages
b) Tabular operator variable display/entry pages
c) Graphical display pages

The status display pages show the value (or state for 2 state digital channels) of up to 60 channels (3 columns of 20 rows). Each channel therefore occupies one third of one line and is presented as follows:-

NNNNNNNNNNNNN VVVVVuuuu

where
N=Name(up to 13 chars),
V=Value(up to 5 chars including decimal point),
u=Units(up to 4 chars).

Digital channel states occupy the 9 character positions otherwise used for Value and Units. The text for digital states are taken

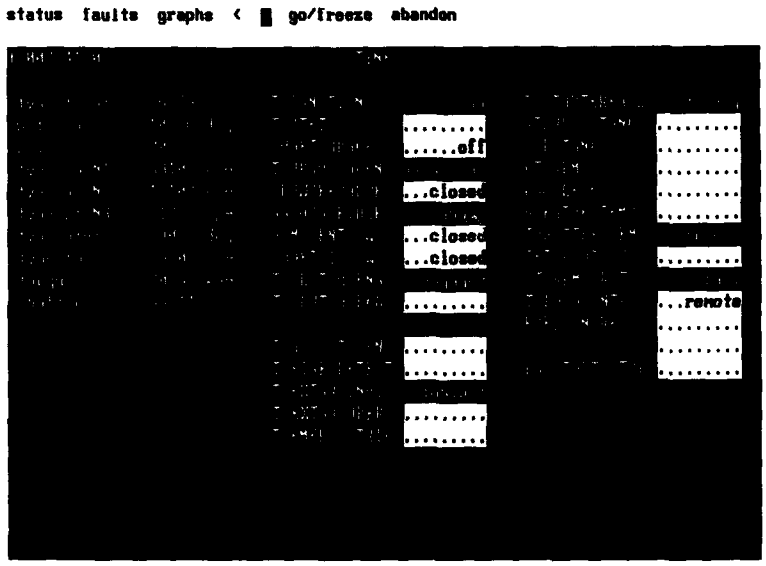


Figure 2 Tabular screen

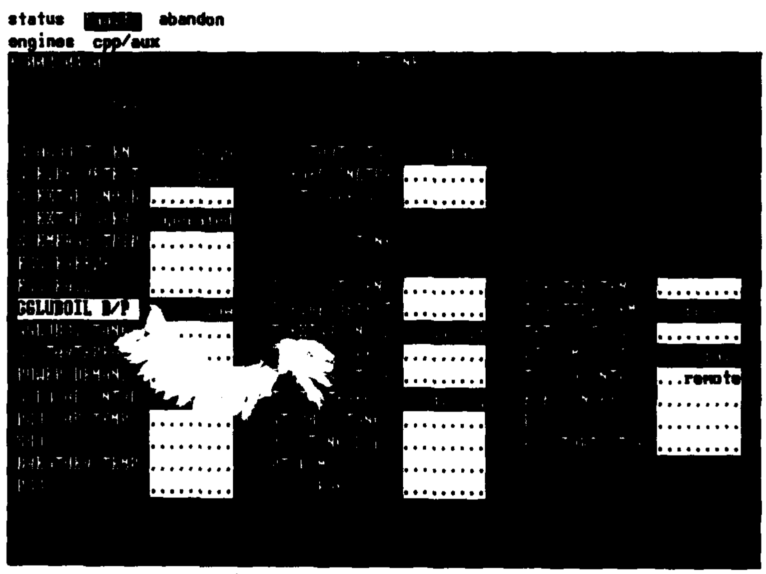


Figure 3 Operator variable screen

from a 'Dictionary' containing pairs of State Description Words (e.g running/stopped etc.).

A sample status display page is shown in figure 2.

The operator variable pages look like the tabular status displays but differ in having a highlighted cursor on one of the displayed variables. In the initial design it was thought that only boolean operator variables would be needed. The operator may change the state of any boolean by moving a cursor to the required variable and pressing the space bar. This will toggle the state of that variable. The operator variables can not be changed during replay.

Figure 3 shows an example of an operator variable screen.

Graphical pages are only available during replay. These pages present time plots for up to 9 channels. Whilst a graphical page is selected, the playback is frozen. The Time Cursor defined below shows the point in the replay which has been reached.

The 9 traces can each be displayed in a different colour. Traces showing the state of digital channels are scaled to fill 7% of the vertical axis but each one occupies a different 10% slice. Traces showing analogue channels have an unrestricted choice of scaling.

The operator may choose one of two configurable time scales, for example one showing a time span of one hour with a resolution of approximately 6 seconds, and the other showing a time span of ten minutes with a resolution of approximately 1 second.

Also shown on this page is a 'Time Cursor' i.e. a vertical line through the graph with marked graduations. This 'Time Cursor' may be moved backwards/forwards using left/right buttons.

The operator may also toggle between displaying names and displaying values for each channel. The values displayed correspond to the channel values at the position of the 'Time Cursor'. Analogue channel values are displayed in engineering units and boolean variable states are displayed as 1 or 0 .

A sample graphical screen is shown in figure 4.

4.4 System Software

An object-orientated design approach was to be used in an attempt to produce systems that comprised of separate independently defined modules that could be 'bolted' together to meet a required simulator system specification.

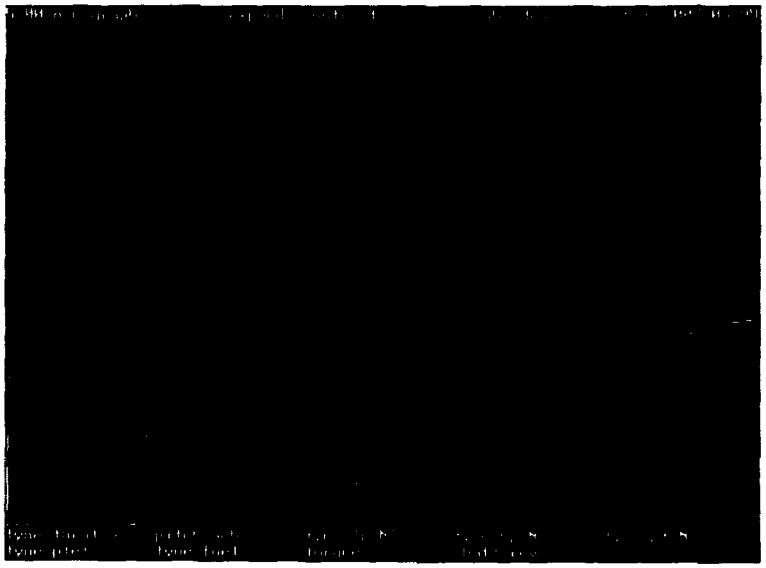


Figure 4 Graphical screen

It was noted that a top-down design methodology would not necessarily yield the desired results since the system architecture evolves from the derivation of the original design specification. If the specification changes significantly (as it will do with requirements for different simulators), the architecture may have to be modified to reflect these changes. The basic architecture should be consistent for all simulator systems.

Since object-orientated design bases the system architecture on characterised classes of data, the most static part of a system, the architecture becomes independent of the processing order of the data which can then be addressed at a later stage. Thus we do not consider the processing order to be part of the inherent system design but as a top-level system implementation for a particular simulation system. (Note that this does not imply that the top-level is to be designed first as it would when using a top-down design methodology; a top-level function has no dependents and is most easily changed within a system without affecting other modules).

The effectiveness of this design methodology and its implementation using the C++ language was to be evaluated during the course of development.

It was further decided that there should be three main

processing tasks:

a) A foreground task in which menu selections are made leading to changes in operational mode and displayed screen.

b) A timed interrupt to command transputers to start simulation execution and to manage data handling i.e. hardware i/o, data recording and data transfer to and from transputers.

c) A background task utilising any remaining time updating the values displayed on the current screen.

From the facility requirements, key data objects were identified that would be required in a general simulator system. For example, the man-machine interface should include screens and displayable i/o data channels; the simulation language interpreter should include symbolic expression tables and schedule definitions. From these general object classifications a hierarchy of classes was derived to implement more specific data types.

Initially, problems were encountered with the identification of classes resulting from the unfamiliar nature of the design concept. It was found that the careful selection of a class was important if the architecture was to be made extendible and reusable. In particular, by making a class too specific, problems could be encountered whenever related classes needed to be created. By identifying common properties amongst sets of data, extendible architecture is produced.

This is well illustrated in the design of the channel class hierarchy. At an abstract level, a channel is defined with just a name. At the next level of refinement, derived classes are defined with the extra features required to model analogue and digital channels. For example a digital channel has a state and a state name pair; an analogue channel has a 12-bit integer value and engineering units scaling. This hierarchy can then be further extended at the appropriate levels to account for channels with different types of values, to impose restrictions or to add more features.

The initial level of abstraction is very important, it is easy to create classes that strictly define the required facilities but which then need constant redesign due to changing requirements. Attention should also be paid to minimising the number of inter-module interactions so that the later stages of design are simplified. The initial hierarchy of classes created is shown in figure 5 and forms the underlying structure of the simulator system.

One particular design requirement considered at this stage, and which influenced the design of most classes, was that any

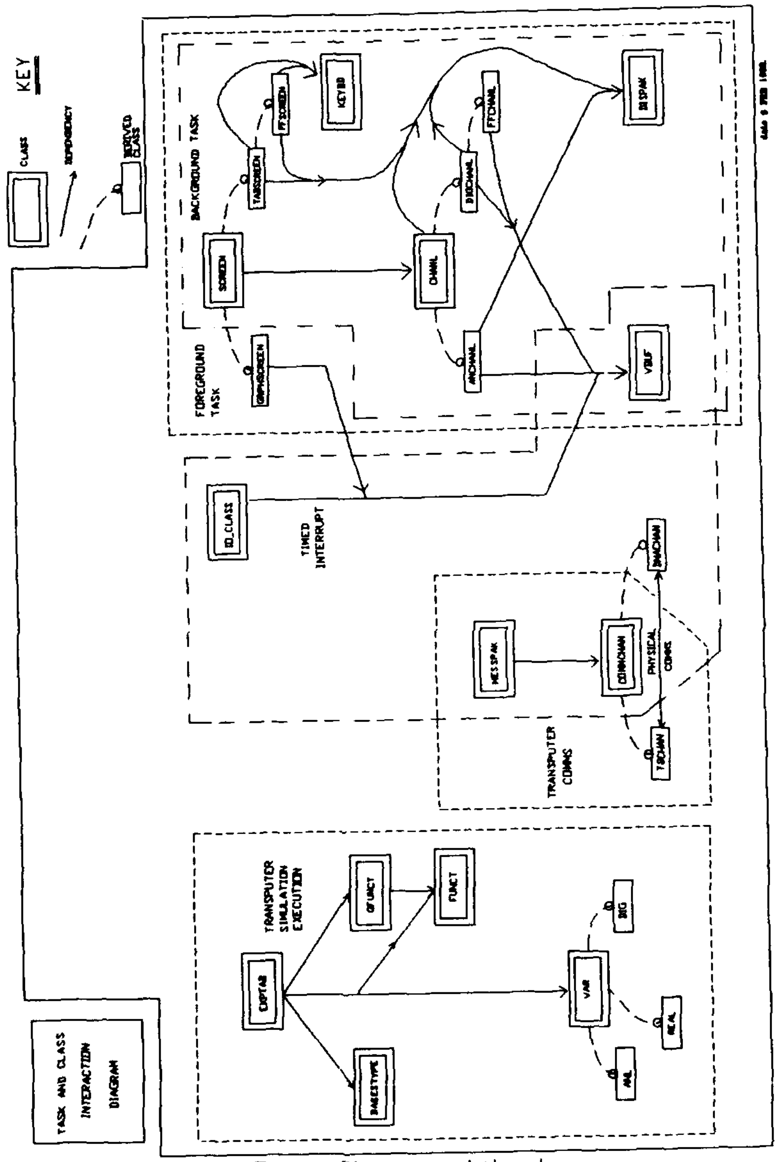


Figure 5 Class architecture

dynamic hardware and simulation data values should be stored in contiguous blocks of memory. This was considered necessary for fast and efficient data recording, hardware i/o and data transfer to and from transputers. The effect of this requirement can be seen in the class hierarchy by the presence of localised buffers (VBUF and VAR) from which most other classes access hardware and other simulation data.

Note that most of the requirements considered at this stage are in fact general facility requirements that will affect the architecture of the system. At this stage we are not considering what we want the system to do, only what it should consist of. This is a very important distinction to make since the operation of different systems will vary to a much greater degree.

After the initial design of a basic architecture, functional aspects can be considered. At this stage we considered the operations required on each class of data and defined the methods that will implement these operations. For example a screen may require configuration, initial static display and dynamic updating. Note that since we have a list of methods attached to a class, it is a simple matter to add extra methods to a class if and as required during implementation.

Although an early decision was made to split the functional operation into three main tasks, it should be noted that this has not affected the system architecture and could have been introduced as a requirement at a much later stage. This is in fact a top-level functional requirement that is independent of any class requirements and which is implemented at the highest level.

5 DEVELOPMENT AND USE

5.1 Algorithm language

The initial experience of using the algorithm language, in particular the lack of any flow control statements, was something of a culture shock for anyone with any experience of programming digital computers. However these were generally people with a control and simulation background and, once the similarity with programming an analogue computer was pointed out, they started to make progress. In such a language what the programmer controls is not the flow of execution but the flow of data.

As experience was gained in programming simulation algorithms we found several advantages to the language, some of which had not been foreseen or planned. We also discovered some real shortcomings.

a) Language shortcomings

Most of the shortcomings related to the omission of any

facilities enabling the algorithm writers to modularise their code. With hindsight this was indefensible. On reviewing this situation it was decided that two mechanisms were needed.

The first was the facility to partition the simulation algorithm i.e. the ability to put some subset of the whole plant simulation into its own file, develop and debug it, and then leave it alone. This was quickly achieved by extending the "algorithm language preprocessor" to accept multiple source files. At the same time a cross reference facility was added to help the programmer keep control of the interfaces between "modules".

The second missing mechanism was the equivalent of the classical subroutine i.e. the ability to reuse one piece of code design multiple times on different instances of data. A typical example which arose in practice is the case of a hydraulically operated valve, of which there may easily be a hundred instances in say a fuel distribution system. To model the state and movement of such a valve takes 5 equations in the algorithm language. We could possibly have regarded this as an instance of the need for a new operator or function in the language but this was not thought appropriate. That technique had been planned for instances where the methods available to the algorithm writer would be too inefficient at run time. It was clear cases like the hydraulic valve would be numerous and we felt that the algorithm writers should be able to define their own "routines" for such cases.

The solution we adopted was to give the algorithm writer a "macro" facility. These macros have formal input and output parameters and also have local internal variables. The local variables are automatically made to be unique Internal variables for each invocation of the macro. To keep things clean and tidy the writer is encouraged to keep macro definitions in separate files by adding an "include file" feature in source files. Incidentally these two facilities (macro & include) were added with absolutely minimal effort by writing a trivial program to do some keyword substitutions in the source and then passing the resulting text through the macro preprocessing stage of a C compiler prior to processing by the main algorithm language processor. This use of multiple programs to "compile" the algorithms is invisible to the user as they are all automated using a "make" control program to build the runable program.

b) <u>The network problem</u>

An instance of the need for a specialised function in order to achieve adequate execution speed did occur in a rather unexpected area. We first came across this in modelling the fuel transfer system in an operator training type of simulator. The model did not need to be very accurate in predicting precise flow rates but it did need to correctly model the paths in which flow occured. Given a typical system, with some 50 or so valves and a

similar number of Tee connections in the pipe system, it is as good as theoretically impossible to write the millions of boolean equations representing the possible flow routes. The same difficulty arises in modelling any extensive "flow network". Other examples occuring in modelling total ship machinery systems are the high presssure sea water system and the electrical distribution system.

The solution adopted is described in terms of a fluid flow system as a specific example is easier to understand. The actual implementation uses a central "route finding" procedure called by different shells providing the differing interfaces to the rest of the model required for fluid transfer systems, cooling systems, and electrical distribution systems.

The Network simulation addition to the simulation language is intended to permit the representation of the system in a manner close to the physical system and to generate a logically correct response to the millions of permutations of valves open/closed tanks full/empty pumps running/not running rather than an accurate quantitative evaluation of all flows. The simulation makes use of a network description table having an entry for each system component. The entry identifies the individual component, the type of component and the other components to which this one directly connects. The different types of system component are:- PUMPS, VALVES, NON-RETURN-VALVES, TEE-JUNCTIONS, TANKS and a CAP. Each component has a predefined number of *network connections* which represent the pipes. Note there is not a pipe component.

The rules used in the simulated network for determining flows are as follows. Flow occurs provided that there is an open route to and from an operating pump. An open route is either a circular path connecting a pump output to the same pumps input or a path between a source of fluid via an operating pump to a sink of fluid. If flow occurs then its magnitude through the pump is the "nominal" flow for that pump. If a circualar path around the pump exists then the pump will not directly influence the levels in any tanks (However the the Gravity Balance side effect of a pump described later). For non-circular paths where branching of flow occurs then the simulation assumes equal division of the flow of a pump amongst the sources feeding it and equal division of its output amongst the sinks it is feeding. In determining whether an open path exists for flow driven by a particular pump, account is taken of the direction in question for non-return-valves. The simulator deals with the case of several pumps running by evaluating flow for each running pump separately and then summing the effects at each source and each sink. The convention is followed that flow into a sink is positive and that out of a source is negative.

The algorithm used works by taking each pump in turn and

calculating and applying the change in contents of tanks as a result of that flow, over the simulated time interval, represented by the calculation iteration interval. This is repeated until all pumps have been dealt with.

The strategy used to deal with each pump is to start at the pump and check that it is running. If it is then a search is conducted on its input side for open paths, from one or more sources.

This is done by using the connection data attached to each component. The link on the input port of the pump is used to send a question to the upstream component asking *can you source liquid*. If it is a valve which is closed or is non-return valve being asked for flow in the wrong direction it answers no. if it is a tank which is not empty it answers yes. In any other situation the component cannot directly determine the answer so it asks the components on its other ports the same question and then returns the answer it gets. In fact only tee-junctions have more than one other port and they ask the component on one of them the question and when they get an answer then proceed to ask the component on their third port, finally they combine the two answers to form their reply to the original question asked of them. To be more precise the answer has two parts one being a count of how many sources have been found and the other being a list of the identity of those sources.

There is one major flaw in the algorithm as described above and that is that if an open path exists which forms a closed ring (a simple example is two valves in parallel joined at each end by tee-junctions) then the question would be passed round and round the ring and never answered. To avoid this possibility a question identity number (1 to 16 in this implementation) is passed along with the question. Tee-junctions remember question numbers that they are in the process of answering. If they receive a question which they are already in the course of trying to answer then they answer "no" to the later request and do not pass the later request on.

Having got an answer about the number and identity of sources on its input port the pump then asks the component connected to its output port if it can sink fluid. This is handled in just the same way as the question about sourcing.

The pump thus ends up with two numbers and two lists: the number of sources and a list of their identities plus the number of sinks and their identities. With this information it is able to determine if flow should exist and if so to apportion it amongst its sources and amongst its sinks. It is also able to determine from whether there exist input and output routes whether to set its delivery pressure at zero, normal, or high.

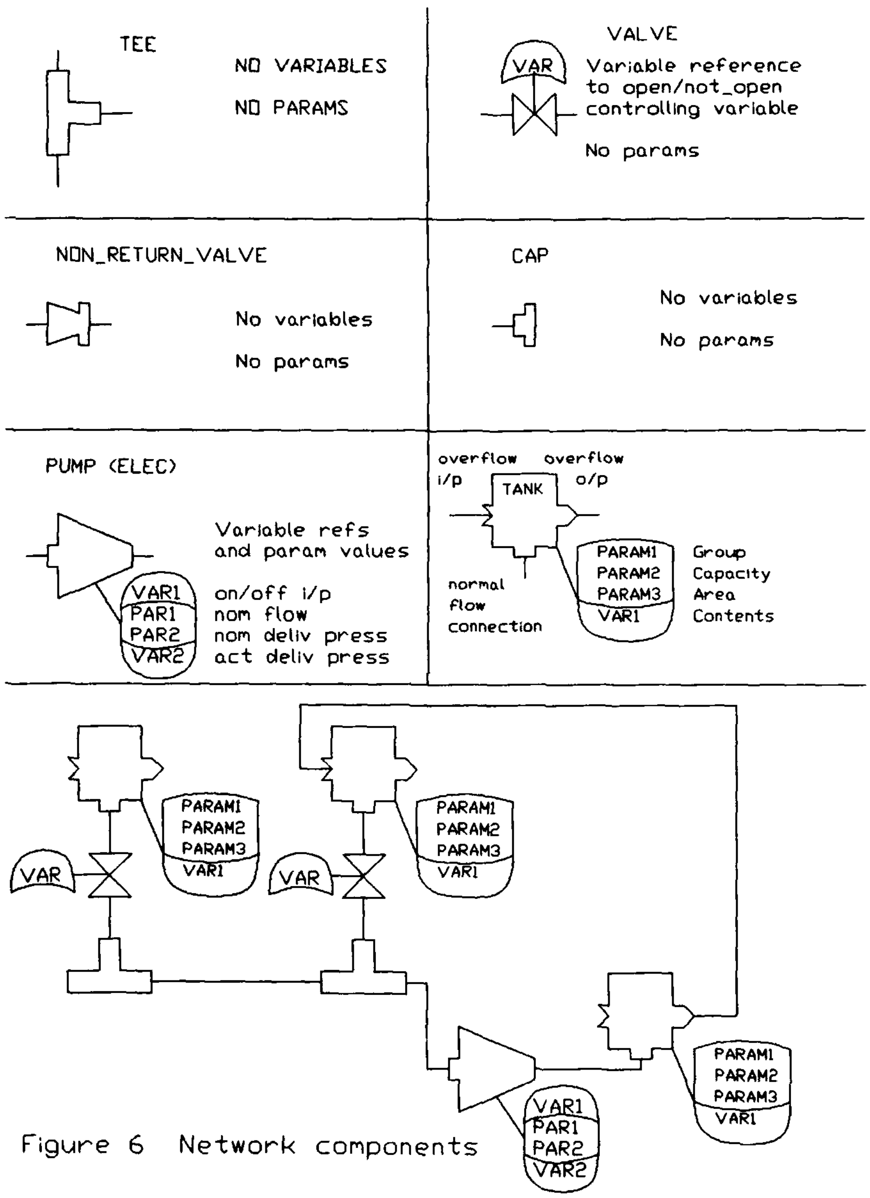


Figure 6 Network components

When used for an electrical network the "can you sink (consume electricity)" questin is launched from each generator. The shell routine for the electrical system uses the answer (number and list of identities of consumers) to totalise the demand and to mark each consumer on the list as having power available. These last power available flags are then used by the individual plant item models .

Figure 6 illustrates the symbols used to diagramatically represent the network components, and the parameters and variables associated with each type of component.

c) Language advantages

Some of the advantages of the algorithm language much appreciated by appliers with previous modelling experience were the freedom from concerns over numeric resolution and overflow, at least inside the model, and the availability of the one and particularly the two input interpolation functions. These they found much easier to readjust as more accurate information about the plant became available than if they had to fit approximating equations to the new data.

With hindsight one of the advantages they reported for use of the language arose out of the lack of flow control constructs that had initially appeared so restrictive. They found that the strict sequential execution of every line of code, on each iteration of the algorithm, lead to a style of coding more likely to produce realistic modelling of the plant in unforeseen situations when compared with case statements for different plant states, or jumping around code once certain conditions are met. Generally they found this pseudo "analogue computing" style also easier to modify and extend when the models performance was found to be lacking in some detail. This seems to arise from a greater similarity of form of the algorithm with that of the plant.

Perhaps the major advantage of using the language was the achievement of one of the principle motivating aims i.e. the use of a SAFE language in which to write the simulation. In spite of our experiences reflecting our worst expectations in terms of late availability of much of the data needed to design and implement the model and its even later correction when the model performance was seen not to reflect some desired characteristic of the plant we had no cases of the dreaded "it all works except that the program 'crashes' once an hour/day/week" attributable to errors in the design or coding of the simulation algorithms.

5.2 Experience of OOD During Project Life Cycle.

a) Ease of Implementation and Maintainability.

During implementation it was found that, due to the highly

modular design with rigidly defined interfaces, we could usually address the implementation of a module without having to lend full consideration to any other, making each module's function relatively easy to understand and implement. This also leads to benefits after initial implementation by producing easily maintainable, self-contained modules.

For instance, a screen need not consider the implementation of a channel that it displays. Any access to channels is directed through its declared public interface and is not dependent upon the implementation of functions provided by this inter face. Thus in changing a screen implementation any changes are localised to screens, no other modules require consideration or modification.

This modular design structure also allows us to build up the system, from the lowest levels, encountering any implementation problems at the earliest stages of development. Top-down functional decomposition can often lead to such problems being encountered at later stages in the project since they inevitably occur at the lowest levels of design. At this stage it can be time consuming and expensive to correct them. By designing and implementing these levels first, any problems are encountered early on and moves can be made to eradicate them.

b) Information Protection.

The implementation of a class with a section devoted to private data ensures that the data attributes of a class can be hidden from other classes. It is then impossible to directly modify an object's private data either intentionally or other wise since access to the data attributes is restricted through the publicly defined method interface. Thus it was found that any errors introduced by modification of a class generally did not propagate into other modules.

Cases in which errors were, however, propagated to other modules were encountered in situations in which a class directly manipulated other class data by overriding the protection mechanism. It was found that such instances can be completely avoided with a more rigorously designed structure. To illustrate this point consider an instance in which this problem occurred:

Several data buffers had to be passed between the transputer nodes and the PC. In order to pass the data over the communications link, a communications class was designed containing transfer methods with parameters for the size of the data block to be transferred and a pointer to its place in memory. To obtain this data, methods that returned this information were added to each class containing data for transmission. However, by obtaining a pointer to a data area, direct access to this data is obtained and we have overridden the protection mechanism. An alternative solution would be to

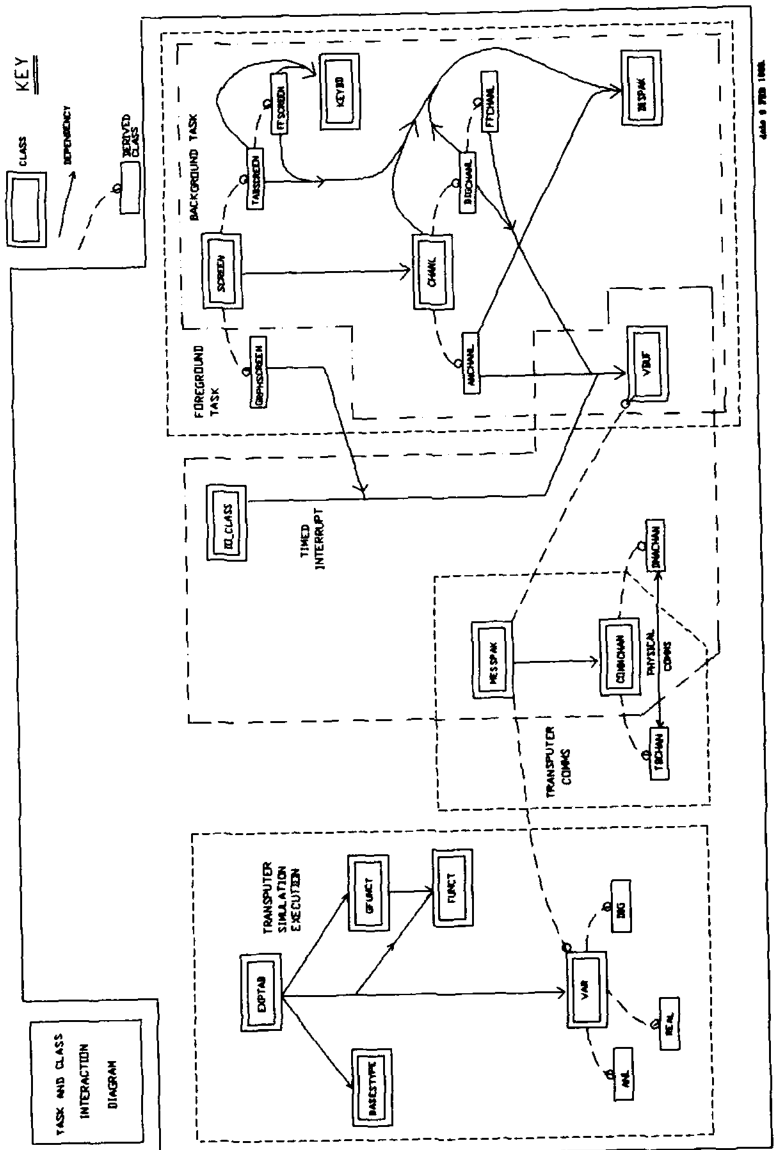


Figure 7 Modified class architecture

inherit the attributes of the communications class into each class that contained data to be transferred across the communications link. For each of these classes the transfer methods can then be called, thus allowing the data to remain private. This is illustrated in the modified class architecture of figure 7. Note that the previous method should not be used in a rigorously designed object-orientated system.

c) Architecture Extendibility.

For our second simulator specification, several extra features were required. For example, operator adjustable analogue variables were required in order to modify analogue inputs to the simulation. These were to be used in a similar way to the existing operator controllable fault injection (boolean variables that are not i/o mapped but that can be changed on the screen to simulate faults in the application).

Implementation of this feature was just a simple matter of deriving the analogue channel properties into a new class for operator controllable analogues with appropriately modified methods. Any methods not redefined in this module apply equally to this module as they do to modules higher in the hierarchy. Additionally, client code at a higher level does not require extension since we have the ability to create constructs of the form 'for whatever type of channel you are, return your value'. This eliminates the use of constructs such as 'if analogue channel type get analogue channel value, if digital channel type ...' which require extension whenever extra modules are added.

d) Execution Time Penalties.

One problem encountered with the implementation of the object design philosophy was that due to the increased emphasis on data structure, it became easy to neglect the functional aspects of the system. Hence in design, many small, trivial class methods were created and typically, for an equivalent functional design methodology, many more functions were implemented. Hence it was found that for our second project which required simulation execution within 0.01s (as opposed to 0.33s), the functional call overhead incurred a run-time penalty and initially we could only execute within 0.1s.

The main source of this overhead was in the design of the symbolic equation table class; it was designed to look like an array with an indexing method that had to be used in conjunction with any data access methods. Hence we had two calls for each variable referenced in each equation evaluation and a number of each of these for the whole equation table.

However, changes were easily made without modification to the system architecture. The data did not require modification and it was a simple matter to create a method to evaluate all equations

in a table with one call.

Note however that execution time can be addressed at the design stage and these problems can be eradicated. As more experience was gained in object-orientated design, the importance of the balance between data and functional design was recognised. Due to the initial use of the completely different design methodology, full consideration of the functional aspect was actually overlooked to some extent.

Incidentally, complementary to the function call run time penalty, we also gain advantages with some method calls. The 'for whatever type of channel you are, return your value' construct is quicker than an 'if analogue channel type get analogue channel value otherwise if digital channel type ...' construct.

e) Module Continuity.

In addition to architecture extension, in some cases module implementations require enhancement to meet new project specifications. However, although for a new project a module may require enhancement, we may not want changes in the module since its implementation has been fixed for previous releases of projects. To overcome this we have the ability to derive properties from the old class into a new one and simply change the implementation of the required sections. Thus we can easily extend modules although we have fixed the implementation in previous versions.

However, we found that due to space limitations in the PC whilst using the PC only version of the program for debugging we could not effectively apply this philosophy since we would be generating more code. Space, in fact, became a problem throughout the latest stages of the project and this can be partly attributed to the fact that many modules had methods that were not used in all projects. This can be overcome with more advanced object-orientated languages and tools such as code optimizers that strip out unreferenced methods at module linkage time.

f) Documentation Difficulties.

Due to the significant influence of defence organisations on our work, all of our standards on software quality and documentation demand documentation structured around approved conventional top down design methodologies such as Mascot. Although we can still, in some respects, structure the textual definitions and descriptions of subsequent levels of design to the required standards, it becomes inefficient. In particular it was found that some documents required specific, conventional diagrams such as data-flow and functional flow diagrams. These will not fulfil their purpose when applied to a object-orientated system. Such diagrams should illustrate design aspects of the system and would not do so for a system designed with a completely

different design methodology. In these cases diagrams, such as the class architecture diagram were developed to meet such criteria meeting the spirit but not the letter of such standards.

5.3 <u>Experience of Using Transputers.</u>

Due to the lack of tools and debugging support we had available for the transputers, the simulation interpreter was initially developed within the PC environment and then ported to a single transputer node. At this stage, some portability difficulties were encountered. This arose because of the method used by current C++ compilers to achieve "type-safe linkage". This relies on the compiler extending function and procedure names to indicate the number and type of parameters expected. Once one is aware of these problems, it is a simple matter to avoid functions that use many long named arguments. Other than this particular problem no significant portability problems were encountered.

Initially, for our first project, a communications protocol was developed to communicate between the PC and one transputer node. Within this scenario it was possible to run the required simulation (twin engines and auxiliaries on a ship) within a 0.1s iteration time. This compares well with a time of just under 1s for execution on a PC. By using a transputer, not only are speed advantages gained, space no longer becomes a problem since each transputer module (TRAM) on a transputer motherboard can be equipped with 2Mb of directly addressable RAM.

For our next project, since we had to simulate a gas turbine at intervals of 10ms with its auxiliaries at intervals of 0.2s, we split the simulation into these two sections so that they could be placed on separate nodes. Thus we required a *communications* protocol to address multiple nodes. It was found that the development of this enhanced protocol was simplified with the ability to split the communications and simulation into separate tasks within a transputer. Communications through a transputer node to another could then occur during execution of the simulation on the node.

It was also a simple matter to reconfigure the system for the two nodes. The interpreter code was duplicated for use on both systems and different application code was built in automatically. All that remained to be set within the code were the data sizes and messages for each node. The transputer network configuration file was then modified to include the connections to the new node. This produced a single file to be loaded to the root of the transputer network which automatically loaded up each transputer throughout the configured network. Hence the PC code did not need be changed to load up multiple nodes.

Since the communications protocol is enhanced to handle a

general network of nodes, it is not difficult to expand the network with this present level of software. All that is required is to set the network definition in the configuration file and the message definitions for each node. If the communication protocol was further developed to include data size bytes within messages, then this process is simplified. Thus, by using transputers, we have the ability to simply split simulation into separate nodes to suit any size of simulation without being limited by execution time.

5.4 Debugging Aids.

During development it was found that, in order for control engineers to test application code and hardware, it became necessary to add some debugging aids that could be accessed from the operator interface during run time sessions. It was possible (though inconvenient) to use existing code debugging tools during initial development in the PC environment. Although they have since become available, such tools were not available for the transputer environment when this work was done. Therefore when the code was ported to the transputers, it became difficult to access the majority of the required data. In particular it became impossible to view floating point values calculated from the expression tables. To this end, a number of debugging aids were gradually added to the system:

a) Ability to inhibit hardware input so that simulation input data could be entered at keyboard.

b) Ability to disable simulation so that hardware outputs could be entered at keyboard.

c) Display of floating point values calculated by simulation.

d) Replay of floating point values.

It was decided that to activate combinations of these facilities, a debug mode number should be entered on the command line; a different system should not have to be created for debugging.

Implementation of keyboard data entry was not difficult since we already had classes of screens that allowed particular channels to be selected and classes of channels that allowed keyboard modification of their values. It was just a matter of declaring extra instances of operator screens on which any channel required for modification could be displayed. To allow channel data to be modified it was possible to build a system in which instances of analogue and digital channels would be declared to be instances of operator adjustable analogues and booleans. However,

rather than having to rebuild in order to add this debugging aid, it was decided to add the required modification methods to the analogue and digital classes and call them only if the appropriate debug mode was set.

The addition of floating point recording and display was a little more difficult. Consideration had to be given to the system execution time and the memory space available. It would have been too time consuming to request and display every floating point value and too much space would be used to record every value over a reasonable length of time.

It was decided that it would be sufficient to select eight floating point values for display at any one time. Additionally, since it would have been impossible to record sufficient data within the PC at the fast rates required, it was decided to record the eight selected floating point values within the transputers. If this data required storage to disk it would not be difficult to transfer it across to the PC block by block for saving to a file. Selection of the floating point values was achieved using eight extra keyboard adjustable analogues.

These debugging options give us a limited ability to test application code and hardware. However they do not allow any fixed simulation data to be modified. To tune any constants and schedules it is necessary to modify application source code and rebuild the system; this can be time consuming. This shortcoming could now be removed by use of the current generation of code debug tools for the trnsputer. This may not be the optimum solution however as the user should not have to learn another operator interface of a different style. Future debugging aids to be added should supply such facilities.

Note that these debugging aids were not designed into the system from the outset. They were added when difficulties in implementing application code were encountered. Although extra classes had to be designed to allow floating point values to be recorded and displayed, these were simply 'bolted on' to the existing architecture without affecting the operation of any other classes.

6 CONCLUSIONS

Generally the development of a simulation framework and toolset has been judged succesful. The subsiduary evaluations carried out within the project have also had favourable outcomes.

The use of multiple transputers to provide generous computing capacity within simulators has been cost effective and fairly easy to achieve. It should be noted however that we restricted their use to a quite simple organisation.

The use of a restricted "safe" language in which to code the simulation algorithms has been successful. However where customer's engineers have to approve the details of the plant sumulation we have found in practice that only diagramatic representations are generally acceptable. Although the simulation language employed in the project is amenable to diagramatic representation, if this is to be a significant proportion of the applications, then automatic compilation from the diagrams should be implemented. The manual translation to textual form is labour intensive and error prone.

The use of Object Oriented design and coding in C++ was also judged succesful and is now the method of choice for the development team. It is not however the solution to all problems. Full attention still has to be paid to all aspects of design. It does however seem to provide a good framework in which to think of the design. It also does seem to encourage the production of code which is more easily reused. The C++ language whilst not a perfect implementation of object oriented designs is quite adequate. In spite of the present lack of an internationally agreed definition of the language the porting of code between two compilers from separate software houses targetting two very different execution processors has been relatively painless.

APPLICATION OF RAPID AUTOMATIC PASSIVE OPTICAL RANGING (RAPOR) TO SHIP CONTROL

by Dr. W.I. Clement and Dr. K.A. Knowles
Department of Weapons and Systems Engineering
United States Naval Academy
Annapolis, Maryland 21402-5025

1. ABSTRACT

Computer processed stereoscopic vision techniques can be used to provide totally passive automatic range and bearing information for a number of ship control applications. These applications include underway replenishment station keeping, restricted passage piloting fixes, buoy location and tracking, and anchor bearing fixes. This paper describes an automated system utilizing a microcomputer and an inexpensive commercial vision system. The image correspondence (range triangulation) problem -- necessary for stereoscopic imaging -- is discussed, along with a technique for simplifying and accelerating the range information processing by using preliminary automatic focus information. Several ship control applications are proposed and discussed.

2. INTRODUCTION

Numerous aspects of ship control could be enhanced or automated if a reliable stereoscopic vision system were available to perform range and bearing calculations to nearby reference points. Range information, via triangulation, is easily extracted from a pair of stereo images if at least one corresponding point in each image is known. Figure 1 shows an overhead view of a pair of cameras in a stereo imaging configuration. With knowledge of the focal length, camera geometry, and corresponding image locations of a given point (say, point "A"), range to this point is easily determined. It is generally agreed, however, that the image correspondence problem is a serious hindrance to the application of stereo vision.

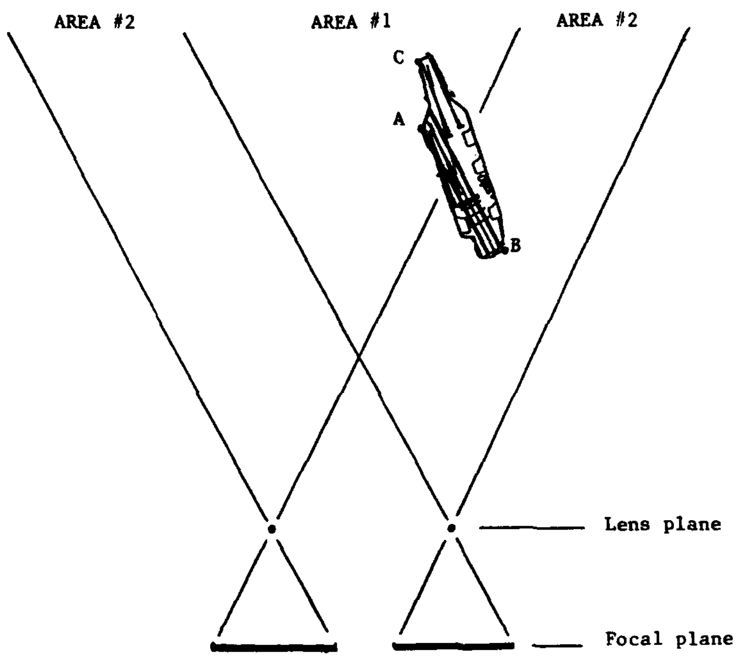


Figure 1. Camera geometry showing field of view.

2.1 Unresolvable correspondence cases

There are several situations in which a solution to the correspondence problem is impossible. In Fig. 1, observe that the field of view of the two cameras is different (accentuated by the small object-range-to-camera-spacing ratio). While there are parts of the scene common to both (area 1), there are, along the edges, objects which only appear in one camera (area 2). This is the source of the first unresolvable case -- the chosen object is not present in both images, as in the case of point "B". This situation is remedied by choosing objects which are within the fringes of the image by an appropriate amount -- an amount which varies inversely with object range. The second problem arises due to the intrinsic nature of stereo images. Due to the parallax between the two images, it is quite possible that an object is visible in one camera, yet occluded in the other.

Point "C" in Fig. 1 is such a point. This problem, too, is exacerbated as image range decreases, since parallax increases. Without specific and detailed knowledge of the scene, it is impossible to completely guard against this situation. However, since this discussion focuses on stereo ranging for ship control, it can be assumed that object distances are relatively large (compared to camera separation), thus decreasing the chance of such errors.

2.2 A proposed correspondence scheme

With a suitably chosen object in one image (e.g., a distinctive edge or point), triangulation for range information can proceed only after the corresponding point is found in the second image. Any algorithm which can locate the best match in the second image will work. However, for real-time applications, this algorithm must be fast. Knowing approximately where to look is one way of accelerating the correspondence solution as well as reducing the possibility of a false correspondence.

The essence of the proposed scheme is to provide the stereo imaging system with a rough guess as to the probable location of the corresponding point in the second image. The key is that, just as correspondence provides range information, range information provides correspondence -- and approximate range information provides approximate correspondence. Therefore, the range triangulation problem can be approached from both perspectives.

A tacit assumption in image analysis is that the camera is focused on objects of interest before one captures the image. However, in order for the stereoscopic imaging system to be autonomous, the camera's focus must first be automated. As a byproduct of this auto-focus subsystem, a first guess at object distance can be made. This knowledge allows for an educated estimate of the displacement of the images between the two cameras (i.e., the correspondence). Finally, standard correlation techniques, performed only on the image area local to the initial correspondence guess, are used to determine exact correspondence, and, hence, accurate target range.

One further point should be made about the use of initial range information to aid in correspondence determination. If the image contains many similar features, such that incorrect correspondence becomes a real issue, the auto-focus system must provide estimated range (i.e., estimated correspondence) with sufficient accuracy as to make the nearest similar feature the correct one. This places a lower bound on the resolution of the auto-focus mechanism -- a bound which is a function of the spacing of similar features in the image.

3. DISCUSSION

3.1 Equipment

The equipment used in these experiments includes two single-camera vision systems (each with vidicon camera, frame-grabber memory board, and cables -- manufactured by BEECO vision systems, Indianapolis, IN), a computer for analysis of the vision data (Zenith Z-248), stepper motors for focusing the cameras, and a parallel I/O board for communication with the motors. In these experiments, only one camera was automatically focused to illustrate the feasibility of the method. Under operating conditions, both cameras would need to be focused, either jointly or independently.

A captured image resides in memory on the vision system's frame-grabber board. For the system used here, the image intensity is spatially sampled to form an array of 256x243 pixels (picture elements), each of which is intensity quantized to one of 128 gray levels (7 bits/pixel). Each pixel resides in one byte of Random Access Memory and the information is stored sequentially, by row.

3.2 Camera focus

An object is in focus when there exists the sharpest intensity gradient across its surface. For digital images, this simply means the largest difference between adjacent pixel intensities, as a function of the lens focus. Since there may be objects at different ranges within a scene, focusing is performed, for these studies, on a small horizontal segment of pixels centered at the image's center. The use of a horizontal line segment, rather than a region of pixels, is due to the camera geometry, as discussed later. The size of this segment is chosen more or less arbitrarily with the intent of making it large enough to encompass distinctive features of the image while, at the same time, being small enough so that, for the most part, all objects within its bounds are at the same range. A satisfactory size can be found by experimentation.

For each position of the focus ring (positions are defined as a given number of steps of the stepper motor), the maximum gradient along the segment of pixels is recorded. As the focal distance steps inward from infinity, this focus process is continued until the gradient has peaked and begun to decrease. After an appropriate decrease threshold, optimal coarse focus is attained by returning the focus ring to the position of maximum gradient. From that point, continuous adjustment of the focal ring, in response to intensity gradient information, provides for improved focus.

Calibration of the lens' focal ring allows correlation between the ring's position and the focal range. This range information determines approximate pixel correspondence between left and right cameras.

3.3 Camera geometry and the correspondence problem

The side-by-side camera arrangement used for these studies imposes a structure on the correspondence problem which simplifies its solution. In the absence of any vertical tilt errors in the mounting, the only position offset between the two images will be in the horizontal direction. Consequently, objects should be chosen which can be easily matched up through the application of horizontal shifts between the two images. This confines the choice of a suitable image to one which shows intensity variation along a horizontal line -- that is, one with vertically-oriented features. This is the reason a horizontal line of pixels was used for the auto-focus routine. For if insufficient detail exists for adequate focus, then it is likely that correspondence will be difficult as well.

The result of our search for correspondence is the lateral separation (measured in pixels) between corresponding points in the two images.

Due to the possibility of vertical misalignment, some modest degree of image continuity must be assumed. That is, these vertically-oriented objects are assumed to extend over enough rows in each image so as to allow the application of matching techniques only on corresponding rows.

As for alignment within the horizontal plane, Fig. 1 may lead one to the conclusion that the cameras must be parallel. At distances which are significantly larger than the camera spacing (the typical arrangement) this is fine. However, as object range decreases, the fringes of the images (area 2 of Fig. 1) grow. If the cameras are angled inward, these fringes can be reduced and even eliminated.

As a result of this additional degree-of-freedom in the camera geometry, a simple calibration procedure is necessary. If an object at known range is studied, it is easy to calculate the pixel separation which corresponds to its range. If it is found that exact correspondence disagrees with this calculation, the difference between the two can be taken as a fixed bias. Later, when approximate correspondence is determined from the auto-focus subsystem, this bias term is added to determine the true starting point in the correspondence search. On board ship, such a calibration procedure could be performed using any object which

is at a known range from the stereo imaging system -- for example, a stanchion or an antenna.

3.4 The choice of features

The choice of a significant feature in the first image for which correspondence is sought in the second is somewhat arbitrary. For these studies, we have chosen the same group of pixels which were used for the auto-focus routine -- if they were satisfactory for focusing the camera then they show sufficient detail to be used for correspondence analysis. All objects within this group of pixels (if, indeed, there is more than one object) are assumed to be at a single range from the camera. That is, the corresponding pixel for each in the group is at the same displacement for all. No attempt is made at object recognition for the selection of reference pixels -- this would be far too time consuming. It is assumed, at this point in the sequence, that the cameras are pointing at the desired object. Furthermore, since focusing is performed on pixels near the center of the image, the chosen points will lie in both images if not occluded.

Due to manufacturing differences in the vidicon cameras and frame-grabber scan rates, there can be significant differences in object widths in the two images. This is a real problem when attempting correspondence -- for one expects to find an exact replica of the reference pixels in the second image. However, when an analog picture (RS-170 standard) is digitized for storage in RAM, each scan line is equally divided into 256 pixels, regardless of actual scan length within the camera. Consequently, if the horizontal scan line of one camera is twice as long as that of the other, all objects will appear to be half as long (when counting pixels) in the first image as compared to the second -- as if one camera had a wide-angle lens and the other did not. This plays havoc on a correspondence scheme and must be corrected. For these studies, the camera which recorded the larger scene (objects were smaller, in pixel count) was scaled, about the center pixel of the row, so that corresponding objects occupied identical pixel widths. The scale factor for this normalization was found experimentally in the lab and remains constant for the given pair of cameras. In actual shipboard implementation, precision CCD cameras would eliminate the need for such calibration.

3.5 The correspondence algorithm

Two algorithms were investigated for determination of exact correspondence -- both based on correlation techniques. If the reference section of pixels from the first image is correlated with those pixels in the second image which lie in the vicinity

of the approximate correspondence, the maximum value for the correlation of the two is the best guess for exact correspondence. The exhaustive nature of an entire line correlation is significantly reduced by knowledge of approximate correspondence -- as is the risk of incorrect correspondence. This is the approach on which we finally settled.

A scheme proposed by Narathong, et. al [1], however, gave initial signs of promise. Under certain conditions, this algorithm can determine exact correspondence after correlation at only a single point. It is based on a Taylor series *approximation* to the intensity gradient and relies on the assumption of slowly varying image intensity. It works surprisingly well even if this assumption is violated (when used in an iterative scheme). However, it was found to be completely unacceptable in cases of large position offset (which should not be the case if the approximate range data is good) and, more importantly, when the average image intensity differs between the two cameras. This latter situation can occur if, for example, the two cameras are set at different *f*-stops. Even though the two images may be identical except that one is an intensity-scaled version of the other, the Narathong algorithm fails. Intensity normalization, a necessary first step to correct this deficiency, is a time-consuming proposition -- making this algorithm less desirable. Even with this normalization, the Narathong algorithm proved less robust than the local correlation technique.

3.6 Range triangulation

Figure 2 is an overhead view of the stereo imaging system, illustrating the camera geometry for range triangulation calculations. Regardless of the lateral position of the object, x, the distance between the two corresponding pixels is found to be

$$(P_r - P_l) = S\frac{D}{(D-F)} \qquad (1)$$

where S is the camera separation, D the object distance from the focal plane, F the focal length, and P_r and P_l are object positions in the right and left cameras, respectively. All that is unknown is the scale factor relating pixel width to linear offset (e.g., K pixels = 1 mm) so that the pixel difference between the two images can be equated to P_r and P_l. This is a constant which is found in the laboratory.

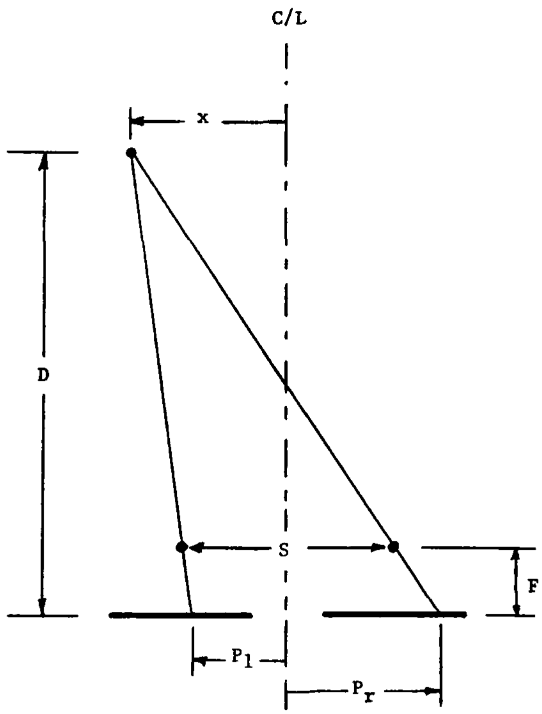


Figure 2. Range triangulation.

The sensitivity of the stereo imaging system, measured as the change in the quantity (P_r-P_l) per unit change in range, D, is found from (1) to be

$$\frac{d(P_r-P_l)}{dD} = -S\frac{F}{(D-F)^2} \tag{2}$$

Since object range is much larger than the focal length, the denominator of (2) may be taken as a constant under nominal conditions. It is clear, then, that sensitivity is directly proportional to camera spacing and focal length.

For example, using K=30 pixels/mm with an F=25mm lens and desiring a sensitivity of 1 pixel change for every 10 feet of range change at a nominal range of D=200 feet, we have

$$\Delta (P_r - P_l) = \frac{(1\ pixel)}{(30\ pixels/mm)} = .0333\,mm \qquad (3)$$

$$\Delta D = 10\,feet = 3048\,mm \qquad (4)$$

which, using (2), yields

$$S = 1624\,mm = 5.33\,ft. \qquad (5)$$

Using an F=50mm lens would halve the camera spacing required for the same sensitivity or, alternately, double the sensitivity. Additionally, from (2), sensitivity changes inversely to the square of the range -- so worst-case sensitivity is at the greatest range where, presumably, it is less important.

The length (P_r-P_l) in (1) can be converted into pixel differences by first subtracting the camera separation, S, (in effect, superimposing the two images) and then scaling by the factor K. For the example above with cameras parallel, the offset between corresponding points would be approximately 20 pixels at range D=200 ft. This is also the size of the image edge fringe within which objects lie in only one camera. Tilting the cameras inward could eliminate this fringe -- providing slightly greater usable image width.

4. A TYPICAL SCENARIO

4.1 Introduction

Several ship control applications suggest themselves for the above-described RAPOR algorithm. In all cases, it is suggested that range (and bearing, if so configured) information obtained be used to assist the Officer Of The Deck (OOD), rather than in an automatic ship control system. If a bearing repeater were

coupled to the camera platform, then both range and bearing could be obtained at the same time.

Several modes of operation could be used, depending upon the requirements of the application. In cases where an occasional single range (and bearing) is desired, a point-and-shoot mode could be utilized. Alternately, continuous range and bearing updates are possible with slight modification to the apparatus and algorithms. For example, a gyro-stabilized platform might be necessary in order for the cameras to remain pointed at a desired target location. Further, providing closed-loop position control of the platform would permit the computer to keep the target object centered in the camera field. This latter control feature could be easily implemented using a stepper motor drive identical to that used for auto-focus, since the computer is already determining object location during each iteration.

4.2 Underway replenishment

Both the point-and-shoot and the automatic continuous range update modes of RAPOR could be useful as an aid to the OOD in maintaining ship separation during underway replenishment (UNREP) evolutions during conditions of emission control (EMCON) restrictions.

During the initial phase of the evolution, the point-and-shoot mode would utilize RAPOR as an electronic stadimeter. The advantage over a conventional stadimeter is that no prior size information about the target object (such as mast height) is necessary and the ranges would be calculated automatically. For this mode of operation, the user (OOD, JOOD, etc.) would point the apparatus so that a vertical object of interest on the other ship, such as a lifeline stanchion or an antenna, would appear in the active region of the video monitor display (that portion containing the highlighted horizontal line along which focusing and stereo correspondence computations are made). Then the computer program would be activated by depressing a push button. The program would quickly focus the cameras, correlate the master and slave images, and display the computed range (and bearing) on the computer CRT. To enhance user confidence, the active region of the video display (the row on which focus computation is being performed, plus several rows above and below) could be updated to show the slave image superimposed upon the master image, much like split-image range finder displays. If corresponding objects match in the active region, confidence is gained in the range computation.

Automatic range updates could be provided during the underway replenishment evolution by adding several refinements. Since both the master and slave cameras possess the same tilt

angle relative to own ship, any slight roll would affect the view of the two cameras equally. So long as a portion of the vertical target object remains across the active horizontal row, the algorithm will still function. Thus, for large ships and fairly calm seas, gyro stabilization may be unnecessary. For small ships and/or rough seas, a stable platform for the cameras is mandatory for continuous operation.

In addition to roll stabilization, some compensation must be made for longitudinal relative motion between the two ships. For small motion, closed-loop azimuth control of the RAPOR platform would be unnecessary. The computer algorithm could be modified to continually shift (horizontally) the active region of the video monitor so as to keep the target object centered in the focus bar. Should sufficient relative longitudinal motion between the two ships occur that the active area (target object) shifts into one of the fringe areas at the side of the display, the computer could sound an audible alarm to signal the user to reposition the cameras to center the object. A similar alarm could indicate that the object has moved vertically out of the active area, perhaps due to roll or a change in list, thus informing the user that reliable range information is not being generated.

For a more robust arrangement in the presence of longitudinal motion, the apparatus could be driven in azimuth. In this manner, when the target object shifted sufficiently far from the center of the image, the computer could reposition the cameras so as to keep the target object centered. With two separate RAPOR units, sufficient information would be available to generate a schematic PPI (plan position indicator) display on one of the computer CRTs to visually depict rough positional relationships between the two ships.

4.3 Anchoring

As with the UNREP scenario, both the point-and-shoot and the automatic continuous range update modes of RAPOR could be useful as aids in the precision anchoring evolution. If the master camera were coupled to a bearing repeater, both bearing and range data could be obtained at the same time, providing the same position fix as two simultaneous visual bearing fixes. In addition, only one reference point would be required, which might prove extremely useful during periods of poor visibility.

Unlike the UNREP operation, large relative motion between own ship and the target object is expected during the anchoring evolution. Therefore, when operating in the continuous mode, the RAPOR system would require automatic azimuth repositioning capabilities to keep the target centered.

Various conditions could be programmed into the algorithm so that an alarm would sound if a critical range (and/or bearing) were to change by an excessive amount. This could be used to detect anchor dragging.

5. CONCLUSION

The Rapid Automatic Passive Optical Ranging (RAPOR) system has the advantage of providing accurate range and bearing information in a completely passive manner. Its simplicity gives it speed and the visual display of its computations provides the user with confidence in the results. In its simplest application, as a point-and-shoot range finder, it does not require knowledge of object size as does a stadimeter. Further, if outfitted with several lenses of different focal length, range and bearing resolution can be tailored to the specific evolution, either manually or automatically.

REFERENCES

[1] C. Narathong, R.M. Inigo, J.F. Donor, and E.S. McVey, "A Target Tracking Algorithm Using a Sensor with Biological Vision Features." Int. J. of Machine Tools Manufacturing, Vol. 28, No. 3, 1988, pp. 217-233.

MINEHUNTER SHIP CONTROL SIMULATIONS

John Woo, NAVSEA
Harold Korves, Unisys
Jeff Neilson, GasTOPS

1.0 ABSTRACT

This paper describes the Minehunter Coastal (MHC-51) Machinery/Ship Control System (M/SCS) man-in-loop (M-I-L) simulation and the results of real-time simulation test runs. The simulation effort sought to assess performance of the MHC-51 closed loop control system.

MHC-51 is a coastal minehunting vessel propelled by twin, aft-located, Voith Schneider cycloidal axis propellers. A diesel engine drives each propeller through a drive train consisting of a variable speed fluid coupling, reduction gear boxes, and shafting. Ship control is effected by a closed loop system consisting of the propulsion control system, ship control displays, the helmsman, the machinery control system, the ship and its environment, and the navigation/command and control *system*. Test results of track-following runs demonstrate that the closed loop ship control system can achieve characteristics similar to that of a second-order linear control system with a damping ratio of about 0.7.

Track following focuses primarily on one steady state condition, namely distance cross track (DCT) equal to zero. The operator may choose different combinations of settings for the two levers and steering wheel to attain this condition. In some test runs, we attempted to achieve this condition by "locking" steering demand to the heading correction value of the previous display update. However, the closed loop control system operating in such a simple steering demand "locked" state may sometimes find a steady state "null" condition, in which DCT is not equal to zero. Results obtained without locking the steering demand, but rather with the operator attaining and maintaining manual control, indicate such undesirable steady state "null" conditions may be eliminated. Further, operator performance varies and depends significantly on his training and ability to maintain concentration on the assigned task. We also conducted test runs to demonstrate positionkeeping/ hovering. These results demonstrate that successful position-keeping/hovering also depends on operator training and performance as well as orientation of the ship to the environment.

2.0 INTRODUCTION

MHC-51 is the lead ship in the US Navy's new Osprey Class Minehunter Coastal Ship program. MHC-51 Machinery/Ship Control System real-time, man-in-loop (M-I-L) simulation tests were performed to assess the performance of the MHC-51 closed loop control system.

MHC-51 is a coastal minehunting vessel propelled by twin, aft-located Voith Schneider vertical axis propellers. A diesel engine drives each propeller through a drive train consisting of a mechanical clutch, variable speed fluid coupling, shafting, and reduction gear boxes.[1] Directional control of the MHC-51 is provided by Voith Schneider propellers capable of producing omni-directional thrust. The propellers give the MHC-51 a high degree of maneuverability, including pure athwartship translational movement, without the use of a rudder and tunnel or azimuthing bow thruster.

There are two MHC-51 propulsion modes: transit and minehunting. In the transit mode, fluid coupling slip is held at minimum value while diesel engine speed and propeller pitch may be changed. In the minehunting mode, diesel engine speed demand is kept constant at 1200 RPM while fluid coupling slip and propeller pitch may be changed.

MHC-51 ship control is effected by a closed loop control system. Figure 1 presents an overview of the M-I-L ship control simulation. Successful performance of ship control operations depends on the performance of the individual components and good or moderate environmental conditions. [2]

The MHC-51 closed loop ship control system starts with the helmsman. The helmsman at the station-in-control ship control console selects one of the two propulsion modes and a ship control display. He monitors the graphics display and also positions two levers, that determine the thrust of each propeller, and a steering wheel, that determines the thrust direction or steering angle of both propellers. The lever and wheel settings are sent to the propulsion control system.

The propulsion control system determines the required values of each diesel engine speed demand, fluid coupling slip, and propeller pitch and steering angle in response to the helmsman's inputs of lever settings, steering wheel setting, and propulsion mode setting. The simulation uses required values (i.e., schedules) identical to those of the operational system. Elements of the propulsion plant are simulated and provide feedback to the propulsion control system (e.g., monitoring current values of the

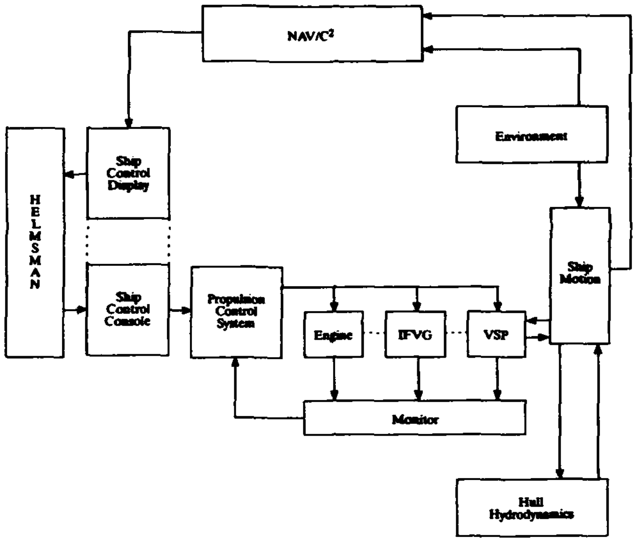


Figure 1. Overview of MHC-51 M-I-L simulation.

actual diesel engine speed, actual fluid coupling slip, and actual propeller pitch and steering angle).

Ultimately, the turning of the propeller results in motion of the ship through the water. Ship motion is also affected by the environment (wind, waves, and current) as well as by hydrodynamic forces on the hull. Propeller performance is also affected by the motion of the ship in the water or, more precisely, the inflow velocity of the water flowing into the propeller blades.

In the operational system, the Navigation/Command and Control (N/C^2) system integrates sensor information from GPS, HYPERFIX, LORAN, the Doppler speed log, the gyrocompass, etc. to estimate position, velocity, heading, etc. In the simulation, the combination of the propeller, environment, and hull/water forces and moments are input to the ship equations of motion. The output of the ship equations of motion yields the position, velocity, and heading of the ship and this is fed through N/C^2.

In both the simulation and the operational system, this position, velocity, and heading information feeds back to the Ship Control Display. By monitoring the display, the helmsman now closes the loop and begins another iteration to effect control of the vessel. As noted, MHC-51 ship control is affected by a closed loop control system. Of course, the system actually consists of several closed loop control systems. For example, each engine

governor contained in the Engine block indicated in the figure provides closed loop control of engine speed. The governor receives a speed demand, monitors the actual engine speed, and, based on the difference between the demanded and actual speeds, determines the amount of change in fuel rack position necessary to bring actual engine speed in line with the desired engine speed.

In the fluid coupling, also called the Integrated Fluid Variator and Gearbox (IFVG), the position of a linear actuator, connected to a scoop tube that drains and fills the coupling, changes based on the difference between actual and demanded actuator position. This demanded actuator position, in turn, changes based on the difference between actual and desired IFVG output speeds (i.e., slip) when the minehunting propulsion mode is selected. Similarly, proportional amplifiers control the lengths of each propeller's longitudinal and lateral hydraulic actuators that position the steering center of each Voith Schneider propeller (VSP) and, consequently, provide propeller pitch. [3, 4, 5, 6, 7]

3.0 SHIP CONTROL DISPLAYS

The M-I-L simulation ship control displays are like those in the MHC-51 operational system. In the operational system, each operator display has an update rate 1 per second. ([1] requires a 2-second update). The M-I-L simulation system operator display has an update rate of approximately 0.6 time per second (an average of 1.7 seconds between updates). The difference in update times is insignificant when assessing performance of the ship control system.

This paper addresses two M-I-L ship control displays: track following and hovering. The operator selects a display via the console menu line pushbuttons located below the display. The operator also uses console pushbuttons to select either of the two propulsion modes.

3.1 Track Following Display

The track following display provides graphic and alpha-numeric information for controlling and minimizing the ship's distance cross track (DCT). A brief description of the M-I-L simulation track following display depicted in Figure 2 follows:

1) On each M-I-L Simulation display page, the page selection made by the operator is highlighted on the menu line at the bottom of the display. The top header line displays the selected propulsion mode (TRANSIT MODE or MINEHUNTING MODE).

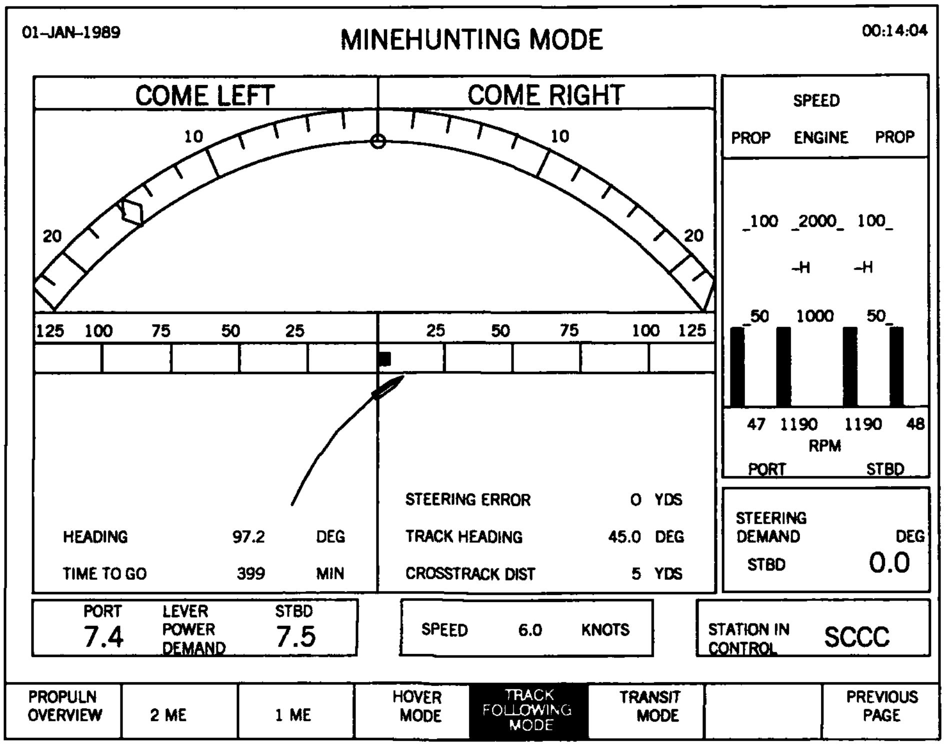


Figure 2. Track following display.

2) Ship speed, positions of the FORE/AFT and PORT/STBD levers, and an indication of which ship station is in control, are provided at the bottom of each display, just above the menu line.

3) Engine speed and propeller RPM, as well as steering wheel demand, are displayed on the right-hand side of each display.

4) Near the center of the display, a portion of the recent ship's track is shown. In this "sea area" portion of the display, the vertical scale is the same as the horizontal scale. The ship's vertical position on the screen remains at the topmost portion of the ship track display and, consequently, the ship track moves down the display generating a so called "waterfall display." The scale changes when the

DCT passes through 100, 400, or 1000 yards. On this display, the ship symbol points in the direction of the ship's heading.

5) Above the track and alphanumeric information, the horizontal bar graph presents a graphic display of the DCT. Zero is located at the center, coincident with the continuously desired track line.

6) A curved display, called the heading correction display, is positioned above the ship's track display. The operator tries to keep DCT equal to zero. The diamond (also called the heading correction symbol) indicates the number of degrees the operator should bring the ship left or right to cause the ship to proceed on the desired track and minimize DCT. Distance cross track equal to zero is maintained when the heading correction symbol (i.e. the diamond) is kept at zero. The operator may choose different combinations of the two levers and the steering wheel to attain this condition. To reduce operator eye fatigue in reading his current steering demand (presented in the lower right part of the display) while trying to follow the diamond, we decided to place a small circle, representing the current steering wheel demand, in the same arc portion containing the diamond.

3.2 Hovering Display

Figure 3 depicts the M-I-L Hovering display. The information at the bottom and right portions of the display are the same as on the track following display. The ship symbol in the main portion of the display points in the direction of the ship's heading. A 3-minute velocity vector, superimposed on the ship symbol, indicates the ship's velocity direction and the estimated distance the ship will travel in 3 minutes.

On the Hovering display, an "X" symbol denotes the location of the target. A danger circle around the target, with a radius provided by N/C², is also displayed. In the future, the position of the Mine Neutralization Vehicle (MNV) will be determined, and its location will be depicted by a diamond on the Hovering display.

4.0 SHIP CONTROL CONSOLE

The M-I-L simulation system ship control console contains functionally equivalent levers, wheel, and pushbuttons. The two thumb-knob-size levers operate individually. The wheel has the same 10 to 1 gear ratio as the operational system. That is, 5 complete turns of the wheel change the steering command from 90° PORT to 90° STBD. Function buttons on a keyboard simulate the operational system pushbuttons for propulsion mode and menu line selection.

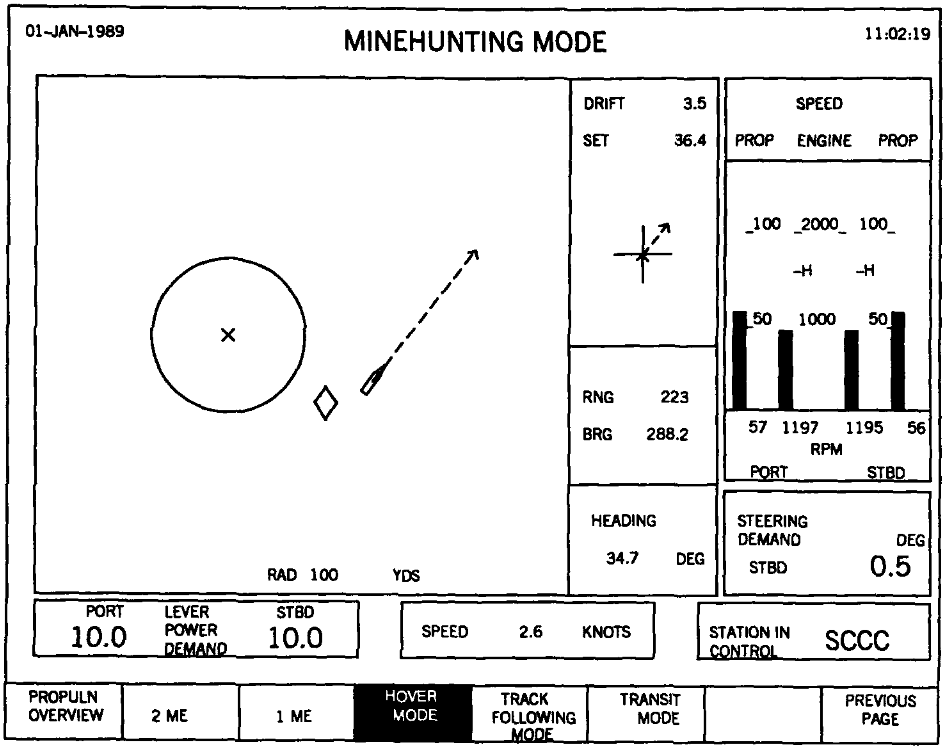
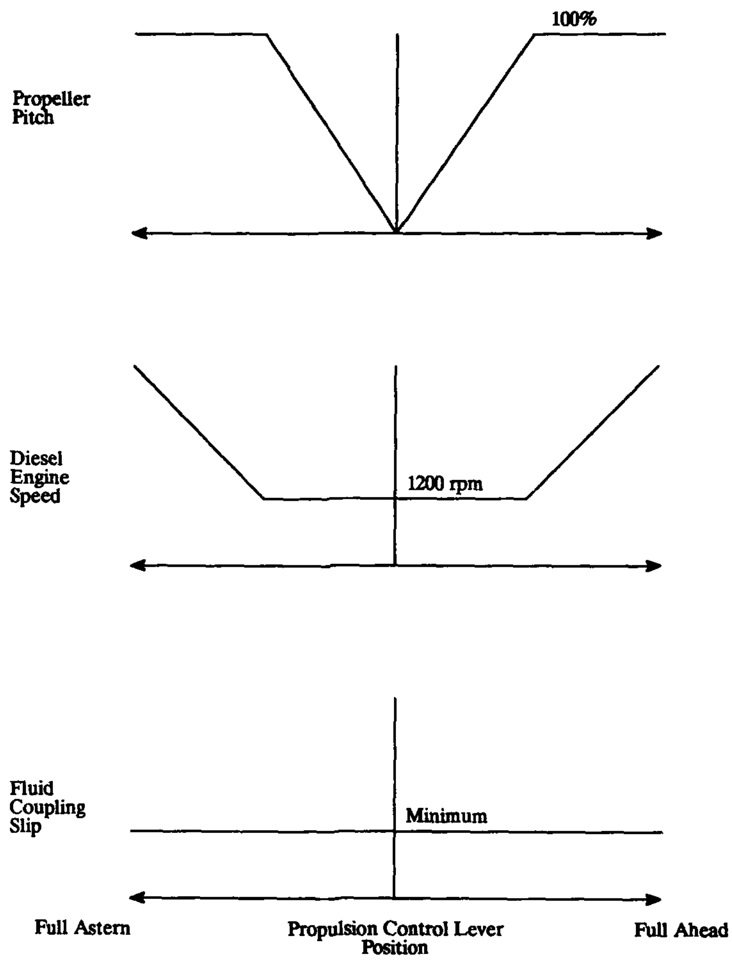


Figure 3. Hover display.

5.0 PROPULSION CONTROL SYSTEM

The propulsion control system software simulates the actual M/SCS including monitoring, propulsion mode selection, scheduling, and diesel overload/overspeed protection functions. The propulsion control system software generates diesel engine speed demands, fluid coupling slip demands, propeller pitch demands, and propeller steering angle demands based upon preset schedules and modified by limiting functions; the limiting functions reflect helmsman commands of propulsion mode, control lever settings, and steering wheel setting. Figures 4 and 5 depict the general form of the propulsion control system schedules for the transit and minehunting propulsion modes, respectively. The schedules have been calibrated to produce linear relationships between propeller thrust and control lever settings.



GR-10-1031-4
eps173 cm 4/26/90

Figure 4. Transit propulsion mode schedule.

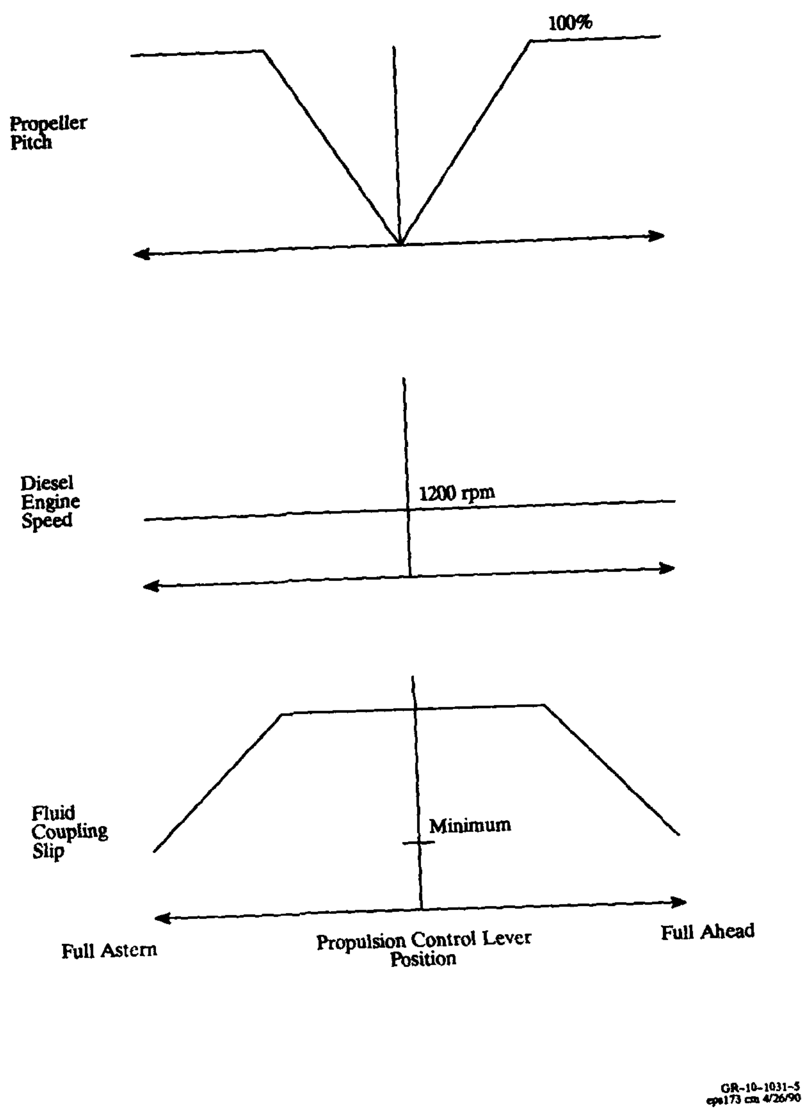


Figure 5. Minehunting propulsion mode schedule.

As noted, in the transit propulsion mode, fluid coupling slip remains constant at its minimum value while diesel engine speed demand and propeller pitch vary. In the minehunting mode, diesel engine speed remains constant at 1200 RPM, while fluid coupling slip and propeller pitch vary. The schedules are constructed to achieve maximum efficiency and ship speed in transit mode and to minimize noise emissions is minehunting mode.

Given diesel engine speed demand, fluid coupling slip demand, and propeller pitch and steering angle demand, the propulsion control system calculates the appropriate signals to be sent to the diesel engine governors, fluid coupling linear actuators, and propeller hydraulic actuator proportional amplifiers.

Hydraulic actuator length demands can be modified by the diesel engine overload protection function. This function limits diesel load via propeller pitch reductions. Similarly, the diesel engine speed demand can be modified by the diesel engine overspeed protection function. This function limits maximum diesel engine speed demand during ahead/astern propeller pitch reversals.

6.0 PROPULSION PLANT

Propulsion plant software simulates the actual MHC-51 propulsion components based on manufacturers' supplied design and performance data. The software provides dynamic representations of the diesel engines and governors, the mechanical clutches, the IFVGs and linear actuators, the reduction gearboxes, and the propellers and hydraulic actuators. The separate torque contributions and inertia values from each drive train component are combined to compute the various drive train shaft accelerations. These accelerations are then integrated to determine the speeds of each drive train shaft segment.

The propulsion plant software also determines the inflow velocities to the propellers so that, in addition to determining the effects of waves on perturbing the ship's motion, the simulation accounts for environmental effects on the propulsion drive train's performance.

7.0 SHIP AND ENVIRONMENT

In the ship motion component, the result of forces and moments acting on the hull determine hull motions. The maneuvering equations are simulated, including longitudinal and lateral motions, plus a third equation for the yaw moment to determine yaw rate. The equations include cross-coupling among the various motions. Contributions to the forces and moment occur because of hull hydrodynamics, the environment, and the propellers [7,8,9].

In the environment component, environmental influences include wind, waves, and current. In the simulation, the wind and wave models generate time-varying forces and moments that act on the hull. The current influences ship velocity with respect to the inertial frame of reference. That is, in the simulation, the ship has an added pure translational motion due to current.

Table 1 defines the magnitude of the good and moderate M-I-L simulation environmental conditions.

Table 1. Specified environmental conditions.

	Moderate	Good
Current (Knots)	1.5	1.
Wind (Knots)	16 - 20	11 - 15
Wave Height (Feet)	6.5	4.9

* Significant wave height, or simply wave height, is four times the RMS of the wave amplitude.

8.0 NAVIGATION/COMMAND AND CONTROL

In the M-I-L simulation system, N/C^2 is modelled to the extent necessary to provide inputs to the displays during transit and minehunting maneuvers. Simulated ship position, velocity, and heading, as well as target position, are available from error-free sensors. Computation of the position of the heading correction diamond symbol is performed based on a combination of five quantities: crosstrack distance, crosstrack velocity, the integral of the crosstrack distance, heading error, and heading rate.

9.0 RESULTS

9.1 Plant Model/Propulsion Control System

Simulation test runs were conducted with the plant model (i.e., propulsion drive train components, hull, environment, and N/C^2) to verify the simulation's implementation and to validate the simulation as representative of the MHC-51 [2]. Having validated the simulation, we used it as a tool for propulsion control system development and testing, and will use it during factory acceptance testing of the operational M/SCS hardware.

We conducted plant model simulation test runs to generate steady state results used to develop the propulsion control system diesel engine speed, fluid coupling slip, and propeller pitch schedules. Incorporating these schedules into the propulsion control system simulation, we performed further simulation test runs to validate the control system algorithms and to tune control system setpoints and gains.

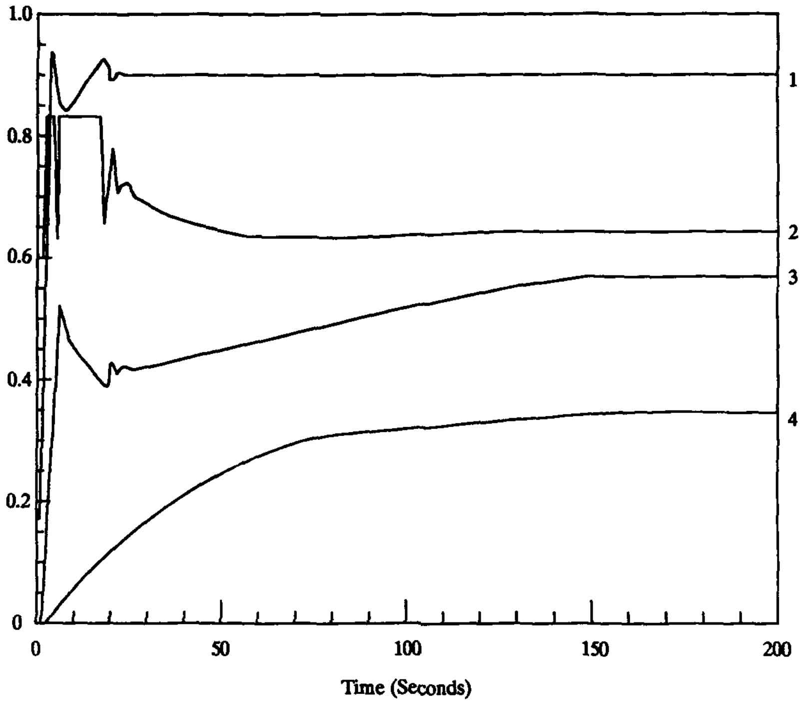
Figure 6 shows simulation test results for a full ahead acceleration maneuver in transit mode. This particular test run was used extensively to tune the diesel engine overload protection function. This function uses a Proportional, Integral, Derivative (PID) controller to calculate propeller pitch reductions as a function of the diesel engine fuel rack error (i.e., actual limit). This test run was used to determine optimum settings for the controller's PID gains and limits, and to optimize the pitch reduction algorithm.

Other test runs covering turning maneuvers, minehunting propulsion mode, and environmental influences confirmed the suitability of the selected values and algorithm. For the maneuver shown, the controller halves the duration of the diesel overload condition that would be experienced with no load control. The performance penalty incurred as a result of the overload control ranges from 0 to about 1.4 knots loss in ship speed during the transient period. However, this penalty makes a negligible difference to the time required for the ship to reach its maximum speed.

Other propulsion control system test runs were conducted to investigate failure modes (e.g., diesel fuel rack to maximum limit, propeller pitch to zero at full load, fluid coupling to minimum slip at idle and maximum slip at full load, etc.), propulsion mode changeovers, and one-/two-engine configuration changeovers.

Lastly, we conducted numerous turning maneuver, turning circle, zig-zag maneuver and asymmetrical maneuver test runs in both propulsion modes, with and without environmental effects, to confirm propulsion control system algorithms and settings and to reconfirm initial ship performance estimates (e.g., time-to-speed, time-to-distance, turning circle radius, etc.). Figures 7 and 8 show simulation test results for a transit mode turning circle maneuver.

The results indicate that, with an approach speed of 10 knots, the ship reaches a steady state turning condition after about 75 seconds, on completion of its first circle. In this steady state condition, the ship has a constant speed of about 2.3 knots and a constant yaw rate of almost exactly 5 degrees per second, producing a turning circle roughly 90 feet in diameter.



LEGEND

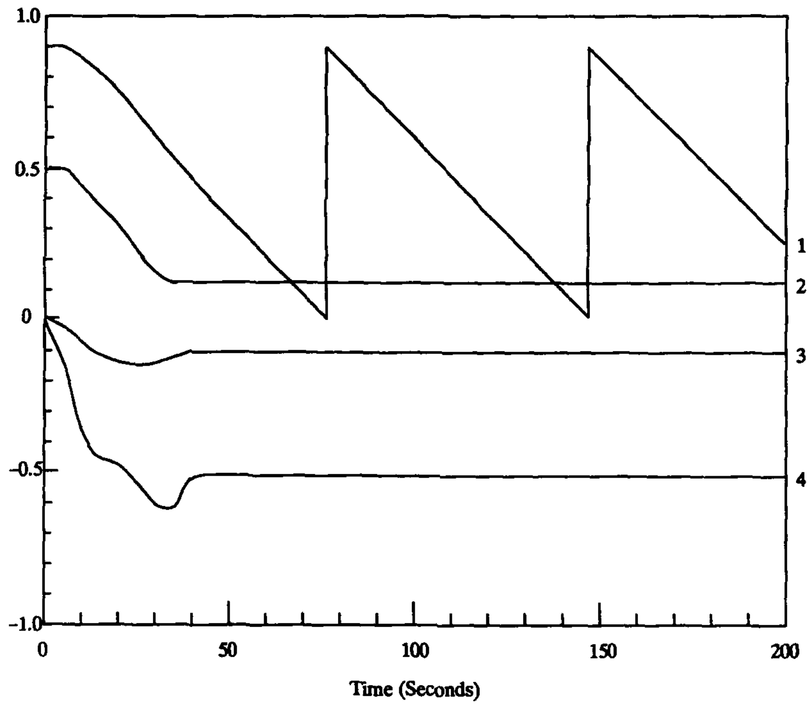
Parameter	Units	Normalization Factor
1. Engine Speed	(RPM)	2000
2. Fuel Rack Pos.	(—)	1.5
3. Pitch Ratio	(—)	1.4
4. Ship Speed	(Knots)	40.0

TEST RUN CONDITIONS

- 0% – 100% Ahead Acceleration
- Transit Propulsion Mode
- Two Engines Enabled
- Sea State 0

GR-10-1031-6
eps173 cm 4/26/90

Figure 6. Full ahead acceleration test run - selected parameters.



LEGEND

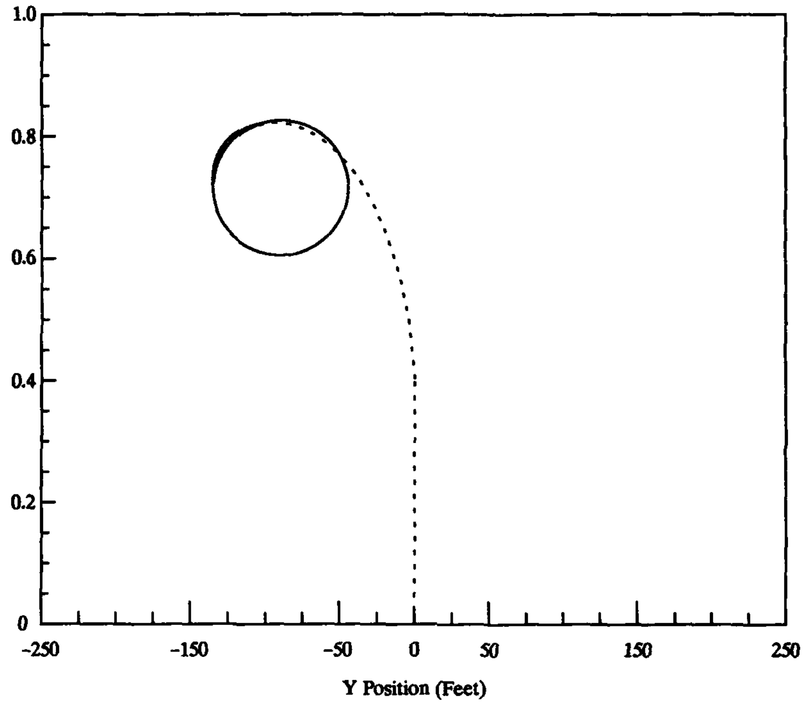
Parameter	Units	Normalization Factor
1. Ship Heading	(deg)	400.0
2. Ship Speed	(Knots)	20.0
3. Inflow Angle	(deg)	180.0
4. Yaw Rate	(deg/sec)	10.0

TEST RUN CONDITIONS

- 10 Knots Approach Speed
- Steering Wheel 45 Deg Port
- Transit Propulsion Mode
- Two Engines Enabled
- Sea State 0

GR-10-1031-7
eps173 cm 4/26/90

Figure 7. Turning circle test run - selected parameters.



LEGEND

Parameter	Units	Normalization Factor
1. X Position	(feet)	389.0

TEST RUN CONDITIONS

- 10 Knots Approach Speed
- Steering Wheel 45 Deg Port
- Transit Propulsion Mode
- Two Engines Enabled
- Sea State 0

Note: Ship position refers to the position of the ship's center of gravity.

GR-10-1031-8
eps173 cm 4/26/90

Figure 8. Turning circle test run - ship track history plot.

9.2 Track Following

We conducted real-time M-I-L simulation track-following test runs in the moderate environment described in table 1. This paper includes a discussion of the results for three representative track following runs.

In the Run No. 1 DCT vs. time plot shown in figure 9, the ship operated in the minehunting propulsion mode with both engines (and IFVG and propellers) enabled. The ship started with a bearing of 90° with respect to the intended track line and 400 yards off track. Throughout the entire run, the two levers were left at the same identical settings corresponding to approximately 4 knots true ship speed for the ship proceeding down the intended track line in the given environmental conditions. The operator sought to drive and maintain the diamond (heading correction symbol) at 0 using only the steering wheel demand.

To highlight performance when the ship is close to and on the track line, figure 9 shows the time history plot for DCT from the time when DCT was -100 yards. By driving the heading correction symbol to zero and getting the heading correction symbol to stay on zero, the operator maintained DCT near zero, except for one excursion of 8 yards.

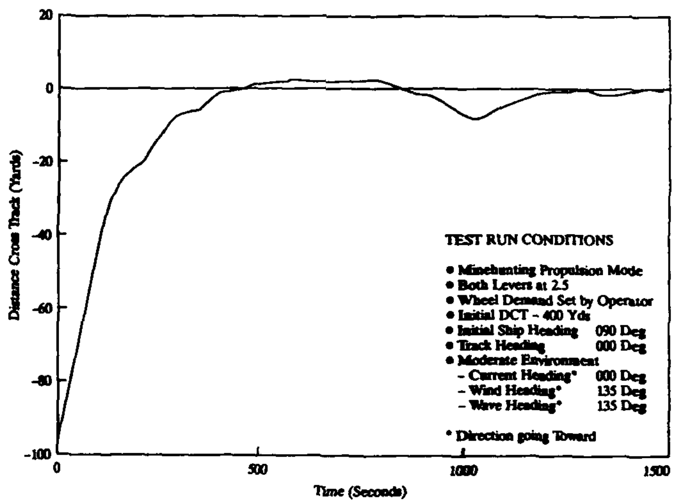


Figure 9. Run no. 1 DCT vs. time plot.

The run with the excursion is shown for two reasons. First, it demonstrates that an operator can get the ship on track and maintain the track under the given conditions. On the other hand, the operator had to continuously concentrate on the task at hand. Specifically, to keep the ship on track in the presence of time-varying wind and waves, the operator had to continuously change the steering demand to maintain the heading correction symbol at 0. When he was distracted, even by a very brief conversation, while simultaneously trying to continue with his task, an excursion such as that shown was likely to occur. This run points out that the performance of the operator depends on both his training and his ability to maintain concentration on the assigned task while avoiding all distractions.

One question arises: What ship control performance may be attained with a trained operator following his superior's directions to get and maintain the track by keeping the heading correction diamond symbol at 0? Different operators may choose to approach and stay on a track using different combinations of the two levers and steering wheel. To assess potential ship performance and the control system and, secondarily, to provide for repeatability of a test run, we took two steps in the conduct of some subsequent track following runs. The two steps involve the demands received from the two levers and steering wheel.

First, we placed both levers at the same setting before commencing a specific test run, and the position of each lever was not altered during the specific test run. We determined the prescribed setting before beginning the run. The value of the setting was such that, after the ship reached the desired track, the ship would proceed down the track in the environmental conditions with a ground speed of about 4 knots while in the minehunting propulsion mode.

Second, simulating an "autopilot" or "helmsman assist" mode, the steering wheel demand was made to follow the heading correction symbol. That is, during each such track following run, we set the steering wheel demand equal to the value of the heading correction on the previous iteration (on average, about 1.7 seconds before the present iteration).

With the procedures just described, each such track following run proved very repeatable. In fact, we saved the initial ship and environmental conditions for each test run in a file. To repeat a test run, the simulation system coordinator only has to input the file, set the levers to the prescribed initial setting given, and start the run simulating the helmsman perfectly following the heading correction symbol.

The Run No. 2 DCT vs. time plot shown in figure 10 is for one such "autopilot" or "helmsman assist" track following test run. In

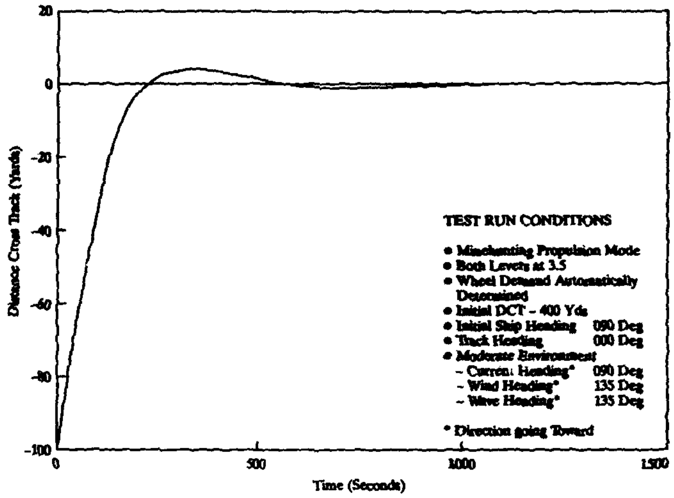


Figure 10. Run no. 2 DCT vs. time plot.

this particular test run, the initial conditions were the same as for Run No. 1 except for the current direction and lever settings. As before, the ship started with a bearing of 90° with respect to the intended track line and 400 yards off track. In the test run results shown here, the first overshoot is 4.3 yards. This test run demonstrates that the closed loop ship control system exhibits characteristics similar to that of a second-order linear control system with a damping ratio of about 0.7. This is shown by considering the first and second overshoots of the system response and using the approximation that the damping ratio may be estimated by minus (1/PI) times the natural logarithm of the absolute value of the second overshoot divided by the first overshoot. As the ship proceeds down the track in the steady state condition, it has a crab angle of about 20 degrees and a ship speed of about 4.1 knots.

Figure 11 shows another test run DCT vs. time plot. In this particular test run, with the same environmental conditions as in Run No. 1, we observed somewhat different results from the other test runs. In this test run, the response does settle in and the ship proceeds down the track in a steady state condition with a ship crab angle and ship speed of about 4 degrees and 4.4 knots, respectively. However, the course-made-good by the ship is about four yards to the left of track.

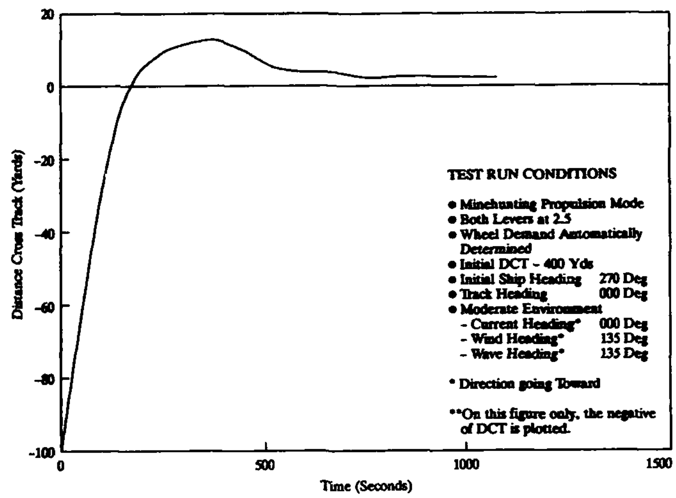


Figure 11. Run no. 3 DCT vs. time plot.

To put the results of this test run in perspective, recall the conduct of the test runs. The steering demand was "locked" to the value of the heading correction *(i.e., steering demand* was automatically set equal to the heading correction value at the previous update). During the test run, a "null" condition was reached where the control system response was steady (i.e., effectively, each of the three quantities, heading rate, course-to-steer minus track heading, and cross track velocity were each equal to zero) but with a non-zero distance cross track. Simultaneously, the heading correction diamond symbol and the steering demand (that was "locked" to the heading correction value) were holding relatively steady to the STBD side (i.e., COME RIGHT) with a value of 1 to 2 degrees. That is, the system was content with a non-zero heading correction diamond value.

We attempted to correct this steady offset by allowing the integral-of-DCT term to become larger. However, as is common in control system studies, this lead to an undesirable underdamped response for the other environmental directions. Since MHC-51 is not required to have an "autopilot" or "helmsman assist" mode, we did not aggressively pursue further modifications of the heading correction algorithm for the "autopilot" or "helmsman assist" mode. Nevertheless, Run No. 2 and many other simulation runs demonstrate the viability of an "autopilot" or "helmsman assist" mode. Implementation would likely involve adaptive control

constants (including the maximum value of the integral term) where the values of the constants depend on the most recent value of the heading correction diamond.

The key point is this: MHC-51 track following is primarily interested in one steady state "null" condition, namely DCT equal to 0. This condition is attained and maintained when the heading correction diamond is kept at 0 on the arc of the track following display. The operator may choose different combinations of the two levers and steering wheel to attain this condition. In the M-I-L simulation autopilot-type test runs, we attempted to achieve this condition by "locking" the steering demand to the heading correction symbol. In fact, this does help to get the ship close to the track and will help to keep the ship on track when it is already on the track. However, the closed loop control system operating in such a simple steering demand "locked" condition may find a steady state "null" condition that is not desirable. This undesirable condition may be characterized by a steady, non-zero value of the heading correction or, sometimes, the heading correction equals the sum of a constant plus a sinusoid. An operator can choose to drive the system such that the heading correction value is always kept near 0.

9.3 Position Keeping/Hovering

The hovering display does not provide any cue or helpful aid similar to the heading correction symbol (and the associated COME RIGHT and COME LEFT) on the track following display. Consequently, positionkeeping/hovering performance depends heavily on the training and performance of the ship control operator and also on that of other ship personnel. Different operators may choose to maintain position and relative bearing to a target using different combinations of the two levers and the steering wheel. Also, during mine countermeasures operations, the CIC subteam receives additional information from the Advanced Minehunting Sonar System AN/SQQ-32, Mine Neutralization System, and N/C^2 systems and subteams. Such additional inputs to the ship control operator and his supervisor impact positionkeeping/ hovering performance but are not included in the M-I-L simulation. With these constraints, we conducted position- keeping/hovering test runs to demonstrate positionkeeping/ hovering. Because of the variability in individual operator performance during positionkeeping/hovering, test runs could not be conducted in a manner providing for exact repeatability of the runs.

We conducted four positionkeeping/hovering real-time M-I-L simulation test runs in the good environment described in Table 1, with the directions of the environment elements identical with those of the third track following test run described above. Specifically, the current comes from the south and the wind and waves come from the northwest.

For each positionkeeping/hovering test run, a target is presumed to be 200 yards forward of the ship's position at the start of the run. At the beginning of the first test run, the ship is on a bearing of 000, with the two levers set initially at 0 and the steering wheel at 0. The ship is in the minehunting propulsion mode with its initial speed and heading rate near 0. With both engines (and IFVG and propellers) enabled, the operator conducts a positionkeeping/hovering run. In each run, the operator tries to maintain both ship heading and position within a 100-yard circle centered at each starting point.

After attempting to maintain position in the first circle for about 10 to 15 minutes, or until the ship goes outside the 100-yard circle, the operator then tries to maintain position near a second positionkeeping/hovering point. In this second run, the operator tries to maintain position with a bearing of 90° and a range of 200 yards, with respect to the target, in another 100-yard circle. After this second run, the operator is tasked with maintaining position near a third positionkeeping/ hovering point. The operator tries to maintain position with a bearing of 180° and a range of 200 yards, with respect to the target, in another 100-yard circle or until the ship goes outside the 100-yard circle. Finally, the operator is tasked with maintaining position near a fourth position keeping/hovering point. The operator tries to maintain position with a bearing of 270° and a range of 200 yards, with respect to the target, in another 100-yard circle.

As the results in Table 2 for the four positionkeeping/ hovering test runs (H1 through H4) demonstrate, successful positionkeeping/hovering depends very much on orientation of the MHC-51 with respect to the environment, as well as the training and performance of the ship control operator.

Table 2. Summary of M-I-L simulation hovering runs (minehunting propulsion mode).

Test Run Number	Bearing To Target At Start (Deg)	Time Within 100 Yards Circle (Min)
H1	0	11.5
H2	90	3.4
H3	180	22.5
H4	270	6.5

In test run H3, with the ship pointed into the current, the operator could keep the ship close to the starting point and maintain heading near 180 degrees. In test run H1, with the current coming from the stern, the operator was less successful in keeping the ship near the starting point and simultaneously maintaining heading near 000 degree. In test runs H2 and H4, with the current coming in the athwartship direction, the operator was even less successful in simultaneously maintaining both ship position and ship heading.

10.0 CONCLUSION

The MHC-51 real-time M-I-L system provides an early look at operation and performance of the closed loop control system consisting of the propulsion control system, ship control displays, the helmsman, the machinery control system, the ship and its environment, and N/C^2. We recognized the need for some refinements in the originally specified operational displays during the M-I-L simulation effort, and incorporated the prescribed changes in both the simulation and operational displays.

Track following test runs show that the MHC-51 ship control system can achieve performance similar to that of a second-order linear control system with a damping ratio of about 0.7. Operator performance depends on both his training and his ability to maintain concentration on the assigned task. Results demonstrate that positionkeeping/hovering performance depends on the training and performance of the operator as well as the orientation of the ship with respect to the environment.

The usefulness of the M-I-L simulation system extends beyond the original simulation effort. First, the simulation software will be used during factory acceptance testing of the operational M/SCS hardware to simulate the ship and the propulsion plant. Second, if unexpected ship control and/or propulsion plant problems arise during shipboard tests or at-sea trials, such problems may be investigated and alternative solutions examined using the M-I-L system. Third, the M-I-L system may be an alternative for providing inexpensive operator training in ship control applications and in maximizing the potential benefits of the omni-directional cycloidal propellers.

11.0 ABBREVIATIONS

CIC	Combat Information Center
DCT	Distance Cross Track
IFVG	Integrated Fluid Variator Gearbox
MHC	Minehunter Coastal
M-I-L	Man-In-Loop
MNV	Mine Neutralization Vehicle
M/SCS	Machinery/Ship Control System
N/C^2	Navigation/Command and Control
RPM	Revolutions Per Minute
SCCC	Ship Control Console, Combat Information Center
VSP	Voith Schneider Propeller

REFERENCES

(1) Minehunter Coastal Ship MHC-51 Ship Specification.

(2) Technical Report: Propulsion Control System Dynamic Analysis for the Coastal Minehunter Ship (MHC-51) Machinery/Ship Control System, December 1989.

(3) Voith, "Voith-Schneider Propulsion (VSP)", ts 2142 e, September 1988.

(4) Voith Letter (with Attachments) to Intermarine USA, "MHC 51-MCS Dynamic Analysis and IPLS Simulation Study", 19 July 1988.

(5) Isotta Fraschini, MHC-51 Engine and IFVG Data.

(6) Woodward Governor Company, "MHC-51 Propulsion System Torsional Vibration Report Closed Loop Analysis", 15 June 1988.

(7) A.H. Wendel et al, Tracor Hydronautics Technical Report 85003.03-1, "MHC-51 Maneuvering and Station Keeping Simulation", March 1987.

(8) Maritime Research Institute Netherlands, Report No. 06407-2-VT, Ship Model Data.

(9) J. Feldman, DTNSRDC/SPD-0393-09, "DTNSRDC Revised Standard Submarine Equations of Motion," June 1979.

FAST TIME SIMULATION MODELS FOR THE ASSESSMENT OF MANOEUVRING PERFORMANCE

by D. ten Hove*, P.H. Wewerinke*,**, J. Perdok* and C. van der Tak*

* Maritime Research Institute Netherlands (MARIN)
** University of Twente

1. ABSTRACT

The use of ship handling simulators, consisting of a bridge with bridge instruments, an outside view and driven by a mathematical model of ship motions, can be very costly. Therefore, the development of fast-time simulation models was started, supplementary to real-time simulation models. In this paper two fast-time simulation models are discussed.

The first model is an optimal control model that can be used to compute optimal control settings to execute a prescribed manoeuvre.

The second model describes in detail the control by the navigator/helmsman of a ship. It includes basic human processes like perception, attention, information processing, decision making and control.

Both models are described together with some illustrative examples.

2. INTRODUCTION

Ship handling simulators, consisting of a bridge with bridge instruments, an outside view and driven by a mathematical model of ship motions can provide answers to questions like:

- Can a particular vessel safely enter into and/or depart from a port?
- What are the limiting environmental conditions for a particular vessel?
- What are the minimum required dimensions of the approach channels and turning basins?
- Are the existing aids to navigation sufficient in providing position information?
- How many tug boats should be used with a particular vessel in a specific manoeuvre?

However, simulators can be very costly due to the vast number of conditions to be tested and the large number of test runs to be performed for each condition in order to obtain reliable answers. Therefore, it is often more cost-effective to use fast-time simulation models. Especially early in the design stage, models can be used to analyze systematically all relevant factors of the complex navigator-ship system and to provide a rational basis for selecting design alternatives and relevant conditions to be investigated, e.g. with a ship handling simulator.

3. FAST-TIME SIMULATION MODELS - GENERAL

Fast-time simulation models can be used to analyze systematically all relevant factors of the complex ship control system and to answer many design and operational questions. The main components of the ship control system are (shown in Fig. 1):

- The mathematical model of the ship dynamics, including the effects of the control variables (rudder, RPM, tug forces etc.).
- The models of the environmental disturbances, such as wind, current, waves and bottom and bank effects.
- A description of the task to be executed: sailing from A to B, possible with predescribed intermediate positions, heading, speed, etc.

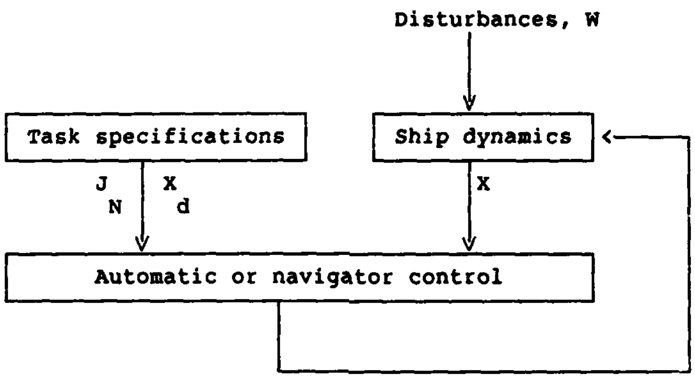


Figure 1. Block diagram of the ship control system

More specifically, the nonlinear, time-varying ship dynamics can be represented by:

$$X(k) = f(X(k-1), U(k-1), W(k-1), k) \tag{3.1}$$

and

$$Y(k) = g(X(k), U(k), k) \tag{3.2}$$

where X(k) is the state vector (consisting of position, speed, heading etc.) at time k, f is a vector function, U is the control vector (comprising rudder, RPM, tug boat forces, etc.), W is the disturbance vector (representing wind, current, waves, etc.), Y is the output vector (indicating the information that is available of the system) and g is a vector function.

The task is defined in terms of a performance measure

$$J_N(U) = E\{ \sum_{i=1}^{N} (X(i) - X_d(i))^T Q_x(i) (X(i) - X_d(i)) + \\ + U^T(i-1) Q_u(i-1) U(i-1)\} \tag{3.3}$$

where J_N is the resulting performance measure corresponding with a given time interval [0,N], which will be minimized by the optimal control U^*, X_d indicates the desired state trajectory and Q_x and Q_u are weightings.

Given the ship dynamics and the disturbances, the task is executed by an automatic optimal controller or by a navigator.

The former is indicated in the following with FORCESIM model, developed by MARIN (Ref. 1). The model basically solves the non-linear tracking problem. The resulting optimal controls represent rudder angle, RPM and external forces, which schematically represent tug boats. As such, FORCESIM can be used to establish upper boundaries of ship manoeuvring capabilities and shows if a desired manoeuvre could possibly be realized.

Ship control by a navigator/helmsman is described by the NAVSIM model, developed by MARIN (Ref. 2). This model deals with the complex visual perception, information processing and decision making and control behaviour of the human navigator.

4. OPTIMAL CONTROL MODEL FORCESIM

4.1 Model and algorithm

FORCESIM can be used to compute optimal control settings, such as rudder angle and RPM, and an optimal use of manoeuvring devices, such as tug boats, for executing a prescribed manoeuvre, as defined

by eq. (3.3), for a given vessel and environment, as described by eq. (3.1), or in continuous time state space description

$$\dot{X}(t) = f(X(t), U(t), t), \quad X(t_0) = X_0 \tag{4.1}$$

and task description

$$J_N(U) = \sum_{i=1}^{N} (X(t_i) - X_d(i))^T Q_x(i) (X(t_i) - X_d(i)) + + \int_0^{t_N} U^T(s) Q_u(s) U(s) ds \tag{4.2}$$

which has to be minimized with respect to U.

Because eqs. (4.1) and (4.2) are coupled, this problem is difficult to handle. However, the optimal control problem can be solved using Pontryagin's maximum principle (Ref. 5).

First define the Hamiltonian

$$H(X(t), U(t), P(t), t) = P^T(t) f(X(t), U(t), t) + + U^T(t) Q_u(t) U(t) \tag{4.3}$$

with $P(t) \in R^n$ a vector function described by the adjunct equations

$$\begin{aligned} \dot{P}(t) &= -P^T(t) \left[\frac{\partial}{\partial X} f(X(t), U(t), t)\right], \quad t \in [t_{i-1}+, t_i] \\ P(t_i) &= P(t_i+) - Q_x(t_i) (X(t_i) - X_d(i)) \\ P(t_N+) &= 0 \end{aligned} \tag{4.4}$$

The maximum principle states that minimizing J_N with respect to U is equivalent to minimizing H with respect to U and and for the optimal control U^*

$$H(X, U^*, P, t) \leq H(X, U, P, t) \tag{4.5}$$

for arbitrary X and P.

The set of eqs. (4.1), (4.3), (4.4) and (4.5) are used to solve the optimal control problem numerically.

In the FORCESIM model a conjugate gradient method (Ref. 6) is used to find an optimal control. This method is described by the following scheme:

1. Solve the state eqs. (4.1) forward in time using a control U_j.
2. Solve the adjunct eqs. (4.4) backwards in time.
3. Find a search direction S_j, improving U_j, using the gradient of the Hamiltonian with respect to U.
4. Compute an optimal step size α^*, such that:

   $J_N(U_j + \alpha^* S_j) \leq J_N (U_j + \alpha S_j)$

   and make $U_{j+1} = U_j + \alpha^* S_j$
5. Repeat step 1 to 4 until the decrease in J_N is small enough.

4.2 Model inputs and outputs

The inputs of the FORCESIM program are the system model of eq. (4.1), describing the ship dynamics, including the effects of tug boats and environmental conditions, a description of the desired manoeuvre $X_d(i)$, i=1, ..., N, and a specification of the weightings Q.

Model outputs are the state trajectory, X(t) (the vessel's position, heading, rate of turn, longitudinal and lateral speed), the corresponding optimal control, U^* (rudder angle, RPM, the external forces which schematically represent the tug boats) and the value of the performance measure $J(U^*)$.

The simulation program provides a method to ascertain the degree of use of the manoeuvring capacity of the vessel involved, as well as the required forces of tugs to keep the vessel within an acceptable track zone and running at the predetermined speed.

In general the FORCESIM program can be used to:

- determine the manoeuvring devices needed for execution of a manoeuvre under various environmental conditions for various vessels;
- select critical simulations and interesting conditions to be investigated, for example, in real-time simulations;
- develop manoeuvring strategies.

In the following paragraph an example of the use of FORCESIM is presented.

4.3 FORCESIM simulation study

As an application of the FORCESIM model an example is discussed of entering an inner harbour through a channel. The example is based on a study for the accessibility of the inner harbour for various types of vessels under various conditions of wind. In this study the FORCESIM program was used to select a limited number of critical conditions to investigate in a real-time simulation study in which experienced pilots are participated.

The manoeuvre considered is an arrival manoeuvre through the channel with courses (sailing inward) of the straight tracks to be followed 280°, 263°, 278° and 345°. The radius of the bends between the straight tracks are:

- course 280°/263° 1/2 n.m.
- course 263°/278° 1/3 n.m.
- course 278°/345° 1/7 n.m.

The desired track and speed information to execute the manoeuvre are determined as in Table 1.

Table 1. Desired track and speed information

Speed at:	Arrival
Course 280°	2 - 2.5 kn
Course 263°	2.5 - 3 kn
Course 278°	3 - 2 kn
Course 245°	2 kn

Since wind plays an important role in the manoeuvring of the vessels, a detailed wind pattern is incorporated into the model. Wind force variation in time is simulated by a Davenport spectrum.

In the channel the bank effects might play an important role. This effect is simulated taking into account the type of vessels and the slope of the borders of the channel. Bank effect is a function of the speed of the vessel and the distance of the nearest border.

The results presented are the simulation results of an arrival manoeuvre with bulkcarriers in ballasted condition under a south-west wind of 25 knots. The main characteristics of the bulkcarriers are given in Table 2.

Table 2. Dimensions of the tested bulkcarriers

LOA (m)	Breadth (m)	Draft ballast (m)
193	30.5	5.5
240	37.9	7.5

Controls available for optimization were rudder and tug boats (in terms of three force components).

The computed trajectory realized by the vessels are shown in Figs. 2 and 3. Sign conventions for the variables are positive speed is ahead (UGR) or to starboard (VGR), positive rate of turn is a starboard turning rate (RGR), positive rudder is to port, positive tug forces are ahead (FX) or to starboard at bow (FYF) and stern (FYS).

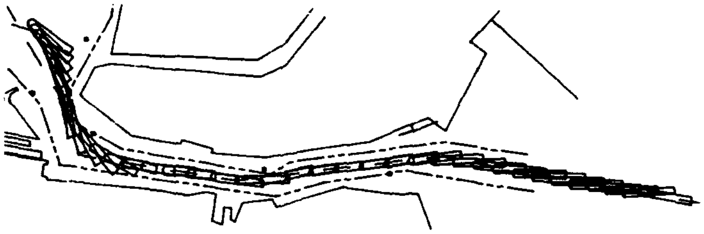
The results show that regarding the required forces it is, from a technical point of view, possible to manoeuvre with vessels up to 240 m length in ballasted condition under the strongest SW wind. However, with maximum wind and size of the vessel it is expected that 100% manoeuvring capacity of the vessel and tugs is required to keep the vessel within the channel limits.

5. NAVIGATOR-SHIP MODEL NAVSIM

5.1 Task and ship model

In order to describe the complete navigator-ship system, a model of the navigator/helmsman has to be included to perform the task defined by eq. (3.3), given the ship dynamics and environment, represented by eqs. (3.1) and (3.2).

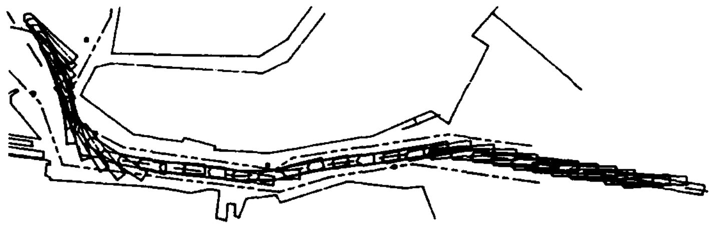
The standard procedure is followed to describe the nonlinear dynamic behaviour of the ship (X) in terms of a state reference (X_0) and deviations (x) from this reference; thus $X = X_0 + x$, etc. This linearization scheme yields a time-varying reference model and the following, time-varying, linear system description



MANOEUVRE : ARRIVAL
SHIP : BULK CARR. 193 m. BALLASTED
WIND : SW 25 kn.

TIME min	RUDD deg	RPM	FX tf	FYF tf	FYS tf	PSI deg	UGR kn	VGR kn	RGR deg/min
.0	35.	25.	1.	-6.	6.	280.	1.9	.0	0.
1.0	-31.	25.	3.	5.	-8.	266.	2.0	.6	-12.
2.0	-4.	25.	-0.	-1.	1.	268.	2.0	.3	7.
3.0	1.	25.	1.	0.	0.	272.	2.1	.3	0.
4.0	0.	25.	0.	0.	0.	271.	2.1	.3	-1.
5.0	2.	25.	1.	0.	-0.	271.	2.2	.3	1.
6.0	1.	25.	1.	1.	-1.	271.	2.2	.3	-0.
7.0	0.	25.	1.	1.	-1.	272.	2.3	.3	2.
8.0	-5.	25.	1.	0.	-0.	274.	2.3	.2	2.
9.0	5.	25.	1.	-0.	0.	275.	2.4	.3	-3.
10.0	-10.	25.	3.	0.	-0.	270.	2.5	.4	-5.
11.0	10.	25.	4.	0.	0.	267.	2.6	.4	-3.
12.0	-23.	25.	3.	0.	-0.	262.	2.7	.3	-3.
13.0	-8.	25.	2.	0.	-0.	261.	2.8	.1	0.
14.0	-6.	25.	1.	0.	0.	262.	2.9	.0	2.
15.0	-2.	25.	-1.	0.	0.	262.	2.9	.1	-1.
16.0	-4.	25.	-0.	0.	0.	261.	2.9	.1	0.
17.0	35.	25.	0.	-1.	1.	261.	2.9	.1	-5.
18.0	-34.	25.	0.	3.	-3.	257.	2.9	.1	10.
19.0	10.	25.	-0.	2.	-3.	269.	2.9	-.2	11.
20.0	35.	25.	-1.	-2.	1.	278.	2.9	-.2	6.
21.0	-30.	25.	-4.	-0.	-0.	277.	2.8	.1	-4.
22.0	10.	25.	-7.	0.	-0.	276.	2.6	.1	1.
23.0	-6.	25.	-7.	0.	-0.	276.	2.4	.1	0.
24.0	-9.	25.	-6.	0.	-0.	278.	2.2	.0	5.
25.0	-33.	25.	-6.	4.	-5.	284.	2.0	-.1	5.
26.0	1.	25.	-3.	1.	-1.	300.	2.0	-.4	23.
27.0	5.	25.	-3.	-1.	1.	319.	1.9	-.5	11.
28.0	0.	25.	-1.	3.	-3.	328.	1.9	-.4	12.
29.0	4.	25.	1.	-1.	1.	343.	2.0	-.5	14.
30.0	16.	25.	-7.	-3.	4.	351.	2.0	-.3	3.
31.0	0.	0.	-4.	-0.	0.	347.	1.8	-.0	-7.
32.0	0.	0.	-4.	-0.	0.	341.	1.6	.1	-5.
33.0	-0.	0.	-4.	0.	0.	336.	1.3	.2	-6.
34.0	0.	0.	-3.	-0.	0.	329.	1.1	.3	-9.
35.0	0.	0.	-5.	-0.	0.	320.	.9	.5	-9.
36.0	0.	0.	-3.	0.	0.	311.	.7	.5	-7.
37.0						306.	.6	.5	-4.

Figure 2. Arrival manoeuvre of the 193 m bulkcarrier



MANOEUVRE : ARRIVAL
SHIP : BULK CARR. 240 m. BALLASTED
WIND : SW 25 kn.

TIME min	RUDD deg	RPM	FX tf	FYF tf	FYS tf	PSI deg	UGR kn	VGR kn	RGR deg/min
.0	2.	25.	-0.	-14.	16.	280.	1.9	.0	0.
1.0	-35.	25.	-4.	8.	-14.	268.	2.0	.5	-12.
2.0	-8.	25.	-6.	-2.	1.	270.	2.0	.2	8.
3.0	1.	25.	-1.	-2.	2.	274.	2.0	.2	-3.
4.0	-10.	25.	-2.	1.	-1.	271.	2.1	.3	0.
5.0	-12.	25.	-3.	1.	-1.	271.	2.2	.3	-1.
6.0	0.	25.	-3.	0.	0.	272.	2.2	.3	1.
7.0	-12.	25.	-2.	2.	-2.	272.	2.3	.3	2.
8.0	7.	25.	-2.	-0.	0.	274.	2.3	.3	1.
9.0	-1.	25.	-1.	0.	0.	274.	2.4	.3	-1.
10.0	-5.	25.	-0.	-1.	1.	271.	2.5	.4	-6.
11.0	7.	25.	1.	-0.	0.	266.	2.6	.4	-5.
12.0	-3.	25.	2.	2.	-3.	260.	2.7	.4	-4.
13.0	-6.	25.	1.	2.	-2.	259.	2.8	.2	2.
14.0	-7.	25.	-1.	0.	-0.	262.	2.9	.0	3.
15.0	-35.	25.	-4.	-0.	-0.	262.	2.9	.1	-4.
16.0	35.	25.	-3.	-1.	1.	262.	2.9	.0	1.
17.0	10.	25.	-5.	-1.	1.	260.	2.9	.2	-5.
18.0	-30.	25.	-2.	6.	-7.	257.	2.9	.0	11.
19.0	21.	25.	-3.	2.	-2.	269.	2.9	-.2	8.
20.0	34.	25.	-6.	-8.	11.	279.	2.9	-.3	12.
21.0	-35.	25.	-12.	-1.	-3.	278.	2.8	.1	-8.
22.0	35.	25.	-15.	-1.	1.	277.	2.6	.1	3.
23.0	-33.	25.	-14.	3.	-8.	275.	2.4	.1	1.
24.0	26.	25.	-15.	-1.	2.	279.	2.2	.0	1.
25.0	-35.	25.	-18.	10.	-10.	283.	2.0	-.1	14.
26.0	8.	25.	-11.	1.	-0.	305.	1.9	-.5	24.
27.0	17.	25.	-7.	-3.	4.	323.	1.8	-.6	9.
28.0	2.	25.	-1.	4.	-4.	330.	1.9	-.4	10.
29.0	-0.	25.	0.	3.	-3.	344.	2.0	-.5	18.
30.0	35.	25.	-16.	-9.	10.	355.	2.0	-.4	2.
31.0	0.	0.	-6.	-1.	1.	349.	1.7	-.0	-11.
32.0	-1.	0.	-7.	1.	-1.	340.	1.6	.1	-6.
33.0	-0.	0.	-7.	1.	-1.	335.	1.3	.2	-3.
34.0	0.	0.	-5.	-1.	1.	332.	1.1	.3	-6.
35.0	1.	0.	-10.	-0.	0.	325.	.9	.4	-7.
36.0	0.	0.	-6.	0.	0.	318.	.7	.5	-6.
37.0						314.	.6	.5	-4.

Figure 3. Arrival manoeuvre of the 240 m bulkcarrier

$$x(k) = \Phi\, x(k-1) + \Psi\, u(k-1) + \Gamma\, w(k-1) \tag{5.1}$$

$$y(k) = H\, x(k) \tag{5.2}$$

where the matrices Φ, Ψ, Γ and H are time-varying (depending on X_0) and w is assumed to be a zero mean, Gaussian, purely random sequence with covariance W.

It is assumed that the navigator derives information about the system from instruments and the outside world. The latter provides information not only about the present state (x) but also about the future desired state (x_d) as explained in Ref. 3. In summary

$$y_e(k) = H_x\, x(k) + H_{x_d}\, x_d(k+N) \tag{5.3}$$

where N indicates the looking time (for a given (nominal) forward speed uniquely related to distance) ahead.

5.2 Model of the navigator/helmsman

The model of the navigator/helmsman (briefly indicated with Human Operator (HO)) comprises various functional elements, which are discussed in the following and shown in Fig. 4.

a. Perception. The HO derives information about the present state of the ship and about the desired state in the near future (by looking forward). The inaccuracy with which these outputs (y_e of eq. (5.3)) are perceived is modelled in terms of observation noise (v_e) which can be related to perceptual thresholds and the level and allocation of attention (Ref. 4). In formula:

$$y_{e_p}(k) = y_e(k) + v_e(k) \tag{5.4}$$

In eq. (5.4) it is assumed that the HO's internal time delays associated with perceptual, central processing, neuromotor pathways and communication and transport delays are negligibly small compared with the system time constants.

b. Information processing. The information perceived by the HO is used to estimate both the present state ($\hat{x}$) and the future desired state ($\hat{x}_d$) of the ship. This estimation process is based on the knowledge of the system dynamics and the disturbance levels. As an illustration, the estimation of the present state is given by

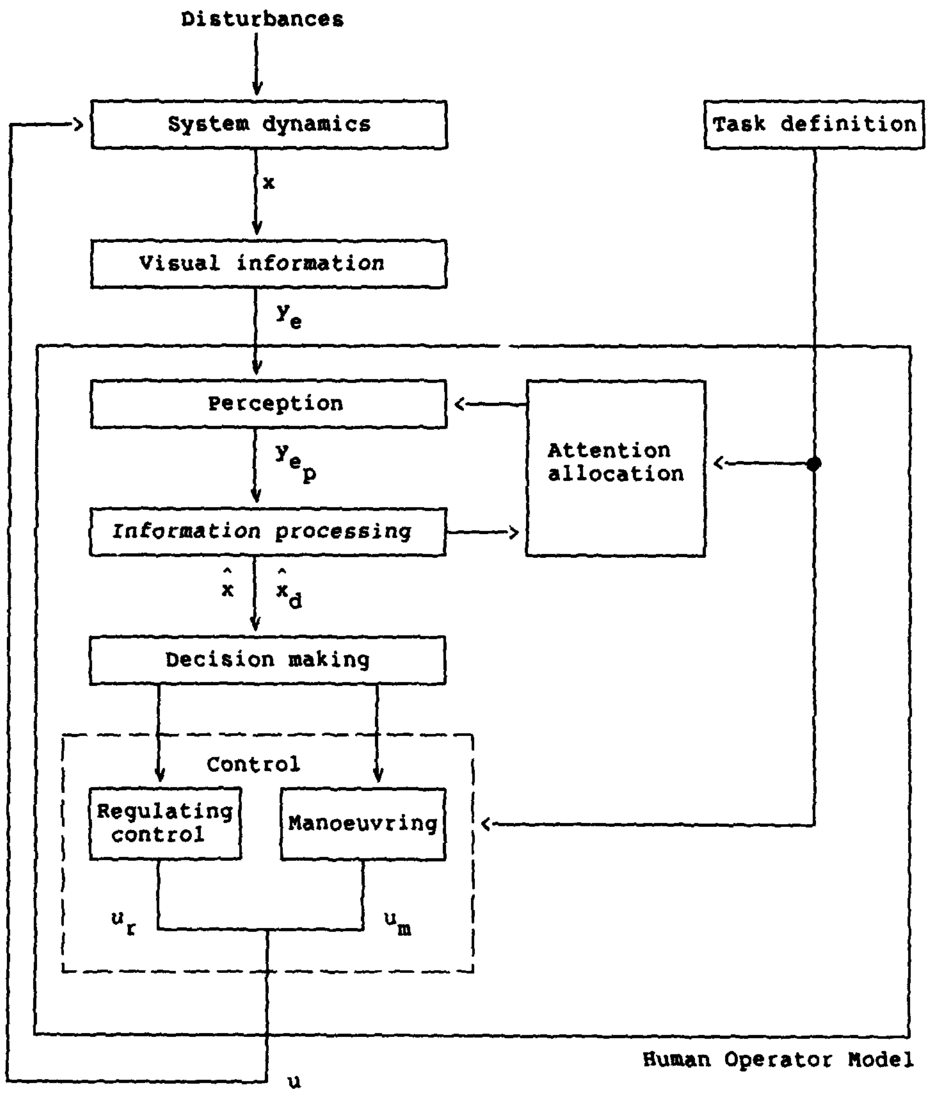


Figure 4. Block diagram NAVSIM model

$$\hat{x}(k) = \Phi\, \hat{x}(k-1) + \Psi\, u(k-1) + K(k)\, n(k) \tag{5.5}$$

with

$$n(k) = y_p(k) - H\, \Phi\, \hat{x}(k-1) \tag{5.6}$$

and

$$K(k) = P(k-)\, H^T\, N^{-1}(k) \tag{5.7}$$

where $P(k-)$ is the covariance of the estimation error $p(k-)$, $(k-)$ indicates the time k before the observation at time k is made, and N is the covariance of the innovation sequence n (i.e. the difference between the perceived information and the corresponding expected value). Eq. (5.5) implies an estimation of the ship behaviour with time and an update of this estimate based on K (an optimal trade-off between the uncertainty about the system state and the reliability of the observations), given in eq. (5.7) and the innovation sequence n, given in eq. (5.6).

A similar process applies to the estimation of x_d. In that case various assumptions can be made about the HO's prior knowledge about x_d and the corresponding internal models. This is discussed in Ref. 3.

c. Sequential decision making. A short term average of the innovation sequence ($\tilde{n}$) can be used to detect systematic changes in the desired state. For example, if a turn has to be executed, this must be envisaged in time so that the proper actions can be planned and executed.

A uniformly most powerful test (generalized likelihood ratio test) can be performed using the recursive expression for the (log of the) likelihood ratio

$$L(k) = L(k-1) + \tfrac{1}{2}\, \tilde{n}^T(k)\, N^{-1}(k)\, \tilde{n}(k) \tag{5.8}$$

This ratio is sequentially compared with a threshold value (corresponding with a given confidence level). Once the ratio exceeds this threshold value, the decision is made at time k_0 that the desired state is changed systematically (with respect to the present reference). Details of this test are given in Ref. 3.

d. Human control behaviour. Once this decision that there is a change in the future desired state is made, an open loop (pre-programmed) manoeuvre (u_m) is planned and executed to achieve the desired future state. In addition, deviations from the present desired state are compensated by means of a closed loop control (u_r) to account for random effects.

The resulting control sequence $u(k)$, $k = 0, 1, \ldots, N-1$ is given by Ref. 3

$$u(k) = u_m(k) + u_r(k) = S(k)\ \hat{x}(k) + S_m(k)\ z(k+1) \tag{5.9}$$

with

$$S(k) = S_m(k)\ W(k+1)\Phi \tag{5.10}$$

$$S_m(k) = -(\Psi^T\ W(k+1) + Q_u)^{-1}\ \Psi^T \tag{5.11}$$

$$W(k) = Q_x(k) + \Phi^T\ W(k+1)\ \Phi_c(k) \tag{5.12}$$

$$\Phi_c(k) = \Phi(k) + \Psi\ S(k) \tag{5.13}$$

and

$$z(k) = \Phi_c^T(k)\ Z(k+1) - Q_x(k)\ \hat{x}_a(k) \tag{5.14}$$

with

$$z(N) = -Q_x(N)\ \hat{x}_d(N) \quad \text{and} \quad k = N-1, \ldots, 0. \tag{5.15}$$

As can be seen from eqs. (5.12) and (5.14), the optimal control utilizes the estimates of the state and the desired state. The latter is involved in the open loop manoeuvre, which is obtained by backwards integration in time of eq. (5.11).

After the manoeuvre (at time $k_0 + N$) the system reference and the small perturbation model are updated according to

$$X_i(k_0+N) = X_{i-1}(k_0+N) + \hat{x}(k_0+N) \tag{5.16}$$

$\phi_i = \phi_i(x_i, u_i)$, etc. (5.17)

e. Visual scanning. In order to describe how the HO allocates his (her) attention among all visual cues, a visual scanning model is derived (Ref. 3) based on the assumption that the HO sequentially observes the visual cues (one at the time) optimally, i.e. minimizing the performance index of eq. (3.3). It can be shown that this implies that the HO is minimizing his total system uncertainty defined as:

$$U(k) = \mathrm{tr}\ [Q(k)\ P_e(k)] \tag{5.18}$$

where Q are given 'weightings' determined by system variables and performance index weightings given in eqs. (5.10)-(5.13).

Using an expression for the effect of one look on this uncertainty U, an (sub)optimal scanning strategy can be derived in terms of the probability of attending to observation i (Ref. 3). The result is:

$$P_i = E\{g_{e_{r_i}}\} / \sum_{i=1}^{NY} E\{g_{e_{r_i}}\} \tag{5.19}$$

with $g_{e_{r_i}}$ an uncertainty measure corresponding with observation i (Ref. 3).

5.3 Model inputs and outputs

The inputs of NAVSIM are the system model of eqs. (3.1) and (3.2), the weightings in the task definition given by eq. (3.3) and the HO parameters: the overall level of attention, the perceptual (or indifference) thresholds and three parameters involved in the decision making model.

The outputs of NAVSIM consists of averages and standard deviations of all state and control variables, probabilities of occurrence, etc. (all statistical information). Furthermore, for many visual informational questions it is useful to establish the information contents of the visual cues (e.g. the effect of a given instrument, a navigation aid and visibility conditions). This can be systematically investigated on the basis of the measure $g_{e_{r_i}}$ (see eq.

(5.19) indicating how navigation performance is affected by a given visual cue. HO measures are available in terms of the optimal allocation of attention (eq. (5.19)) and a workload index.

The model can be applied in port and fairway design, risk analysis, the evaluation of navigational aids and workload studies. In the following, two applications of NAVSIM are presented as illustrative examples.

5.4. Comparison with real-time simulation results

Validating a complex model like NAVSIM requires an ongoing research effort. Two experimental programs that are conducted by MARIN will be discussed in this paper to show the model capability to predict the manoeuvring performance in complex navigation tasks.

a. Manoeuvring a ship into a harbour. NAVSIM was applied to the manual control of an 80,000 DWT bulkcarrier entering a harbour. The manoeuvring situation is shown in Fig. 5.

The ship is approaching the harbour with a speed of 5 knots (part I of the navigation task), using the coast line and the compass (heading) as visual information. A manoeuvre is initiated at a given moment (corresponding with the position of the ship as indicated in the figure) so as to pass the two buoys that provide the navigator with track and heading information. After this manoeuvre the harbour is entered (part III); information about the track is provided by the leading lights at the end of the harbour and by the perception of the harbour entrance. Finally, the terminal position, with zero velocity, must be realized (part IV) based on the visual cues provided by the quay.

The results shown in the figure pertain to phases I and II of the task and concern the ship position and heading. A 99% probability interval of the bow and stern of the ship is also shown. It will be clear from the figure that the probability of hitting the buoys is about 1%. This is partly due to the strong current, for which the navigator has to compensate by means of a relatively large drift angle. Consequently, the task is difficult to perform with an acceptable level of safety.

The experimental results of a simulator program are shown in Fig. 6. The results are based on twelve simulator runs. It turns out that the average trajectory of the experimental runs is very similar to the one predicted by the model. The variability in terms of the 99% probability interval of the bow and stern of the ship is slightly greater than the predicted variability, confirming the poor task performance predicted by the model.

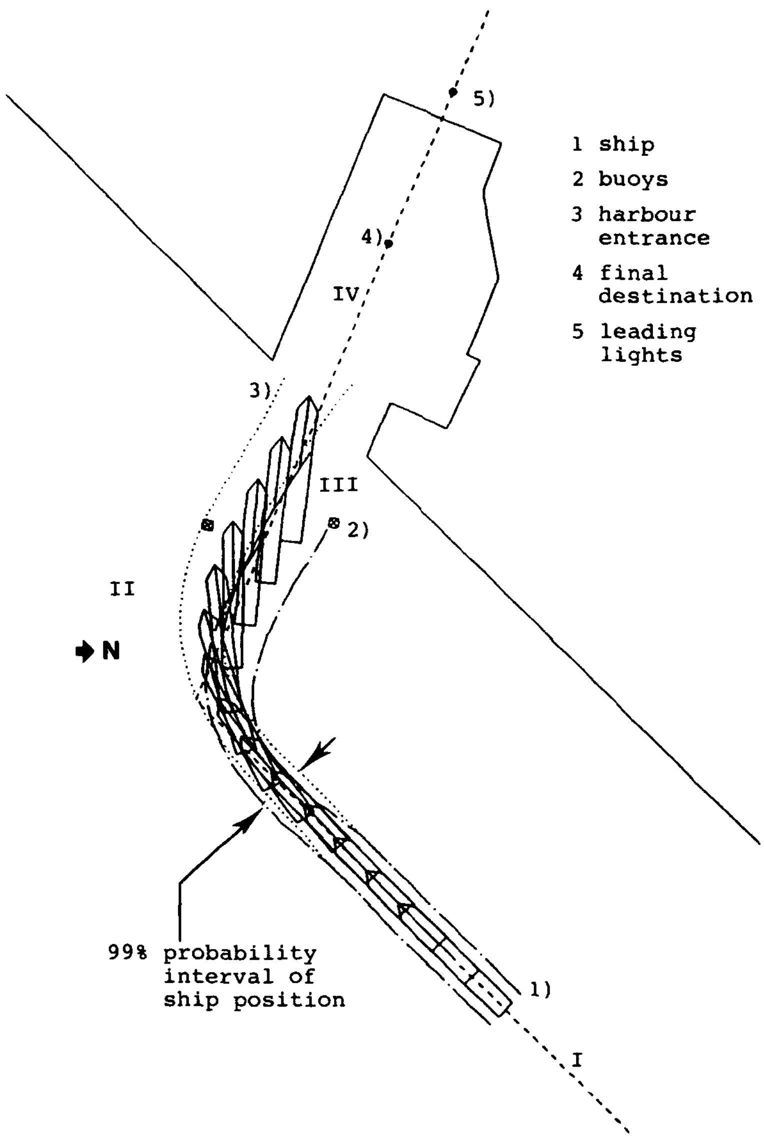


Figure 5. Model predictions of ship control task

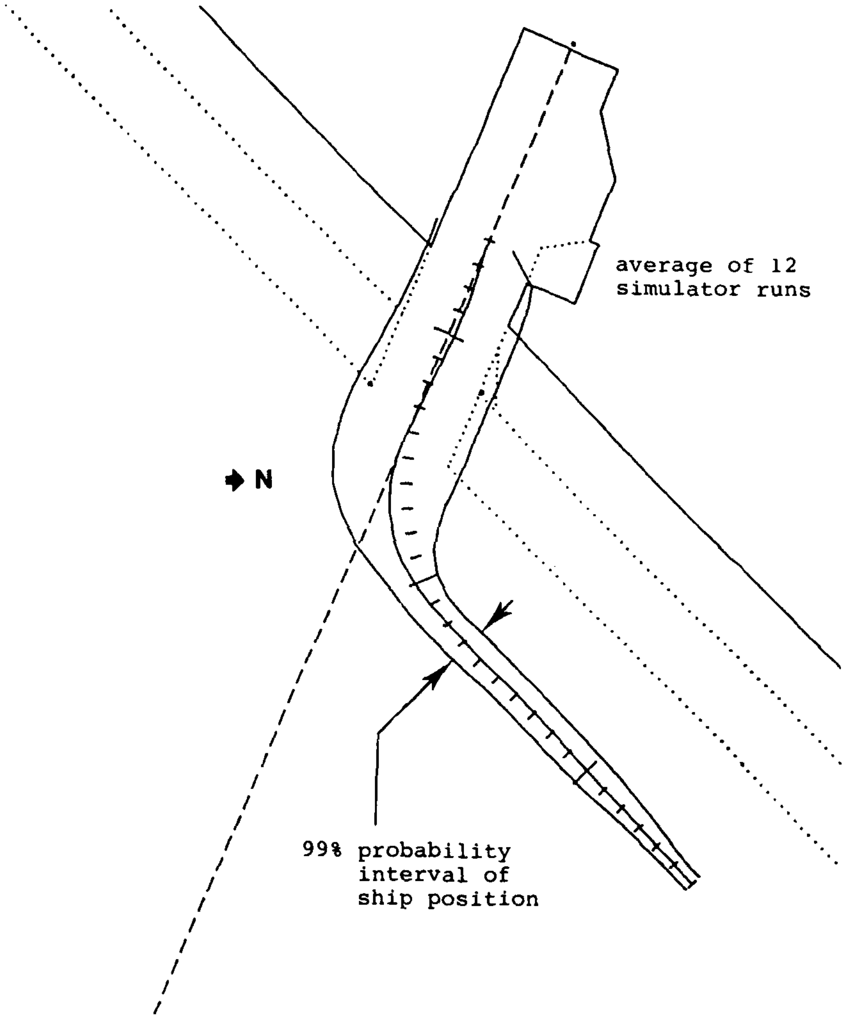


Figure 6. Experimental results of the ship control task

b. Aids to navigation. The second application of the model concerned the effect of aids to navigation (ATN) on safety, ultimately aimed at the assessment of minimal requirements of visual ATN with regard to recognition and identification ranges (Ref. 4).

The task considered is shown in Fig. 7. The total track to be followed by the navigator consists of three trajectories with a total length of 6 nm. The fairway was marked by buoys, located at the way points. The manoeuvres had to be carried out with a 80,000 DWT ballasted bulkcarrier in cross current of 0.75 m/s. A constant speed of 10 knots through the water had to be maintained. The manoeuvres were carried out by four experienced pilots on the manoeuvring simulator of MARIN, equipped with a Computer Generated Image.

Two conditions were investigated. In one condition, only outside view position information was available to the navigator (visual condition). In the other condition, the navigator had to rely on his radar position information (radar condition). Each condition was performed eight times. The simulated manoeuvres were analyzed by computing the lines which have a probability of exceedance of 0.5 per cent at each side.

Fig. 8 shows the results of both the experiment and the model, for both conditions. Inspection of the results shows that there is a reasonable agreement between model results and experimental results.

In addition, the model predicted correctly that visual navigation is superior to radar navigation for the task situation considered.

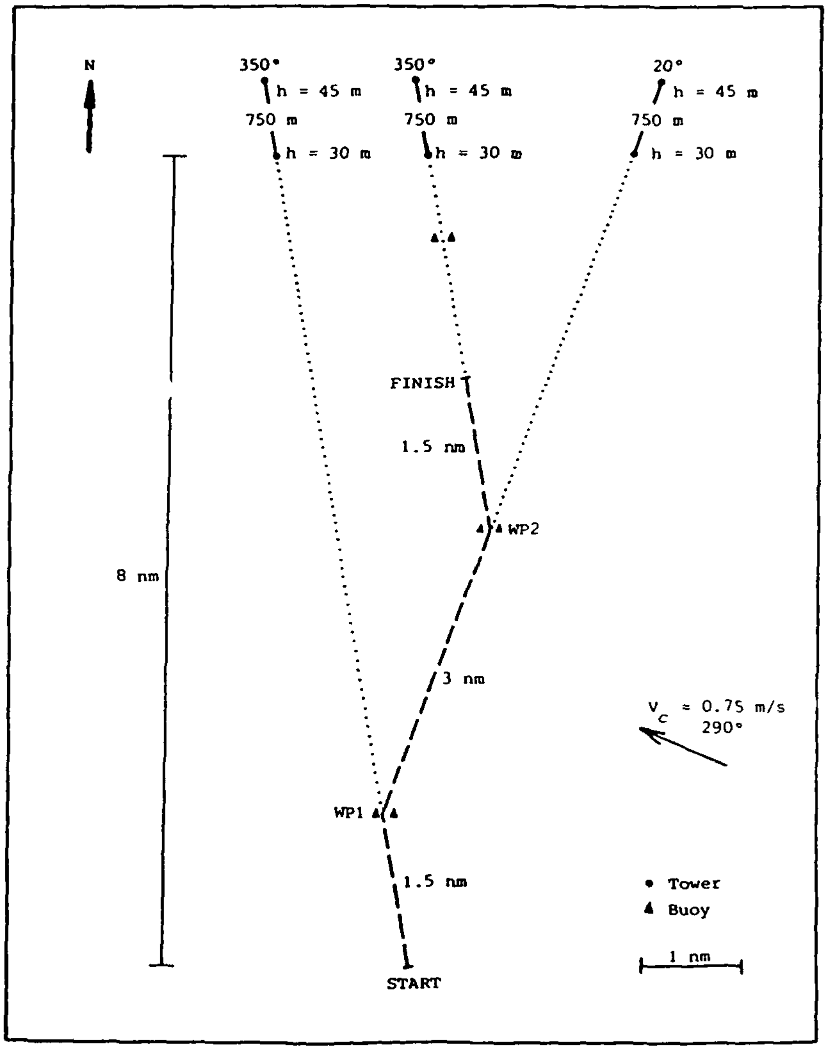


Figure 7. Aids to navigation task

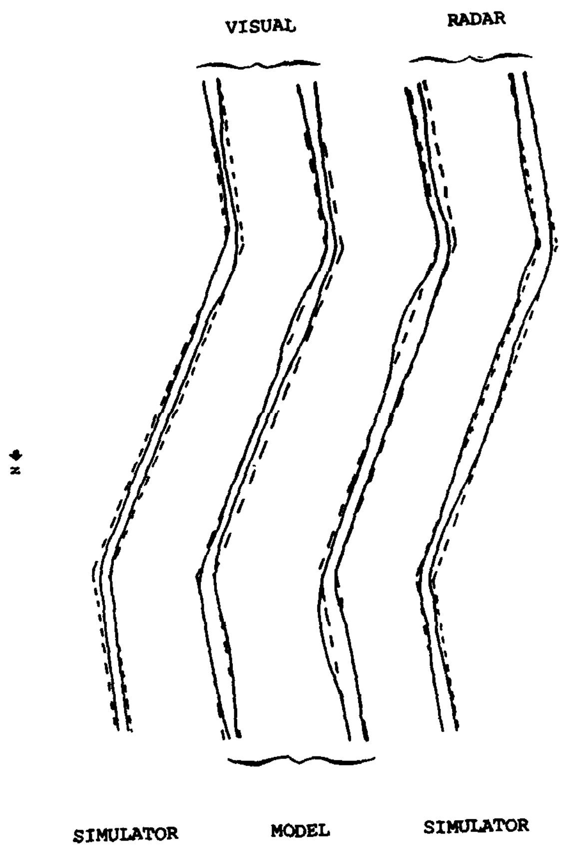


Figure 8. Aids to navigation results - model and experiment

6. CONCLUDING REMARKS

In this paper, two fast-time simulation models are discussed for the assessment of ship manoeuvring performance.

If human factors, like perception and decision making, are considered less important, an optimal control model can be used to analyze the manoeuvres to be performed, without considering the human element. An example of such an approach is the model FORCESIM, which computes optimal control settings, such as rudder angle, RPM and an optimal use of manoeuvring devices such as thrusters and tug boats.

In many operational situations, the human operator plays an important role. In that case, NAVSIM is available involving basic human operator functioning, like perception, attention, information processing, decision making and control.

Two illustrative examples are given comparing model results with real-time simulator data. The results show the model capability to describe the manoeuvring performance in complex navigation tasks.

REFERENCES

(1) J. Perdok, A. Rem and P.H. Wewerinke, "The Use of Real Time and Fast-Time Simulation in Ship Control Studies", IFAC Workshop on Expert Systems and Signal Processing in Marine Automation, Lyngby, Denmark, 1989.

(2) P.H. Wewerinke, J. Perdok and C. van der Tak, "Model of the Human Observer and Controller of a Dynamic System - Theory and Model Application to Ship Handling", Proceedings of the 1988 IEEE SMC Conference, China, 1988.

(3) P.H. Wewerinke, "Models of the Human Observer and Controller of a Dynamic System", Ph.D. Thesis, University of Twente, 1989.

(4) J. Perdok et al., "The Assessment of Operational Requirements of Visual Aids to Navigation using a Model of the Human Navigator", paper presented at the IALA Conference, 1990.

(5) A. Bryson and Y. Ho, "Applied Optimal Control", Wiley and Sons, 1975.

(6) E. Polak, "Computational Methods in Optimization", Academic Press, 1971.